



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR -TE 141599

**PENGATURAN KECEPATAN MOTOR INDUKSI 3 FASA
MENGUNAKAN METODE VEKTOR KONTROL
BERBASIS JARINGAN SYARAF TIRUAN *RADIAL BASIS
FUNCTION* - PD (JST RBF - PD) *HYBRID***

Rizalun Iqbal
NRP 2212106024

Dosen Pembimbing
Dr. Ir. Mochammad Rameli
Ir. Rusdhianto Effendie A.K,MT

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - TE141599

***SPEED CONTROL OF 3 PHASE INDUCTION MOTOR
USING VECTOR CONTROL METHOD BASED RADIAL
BASIS FUNCTION NEURAL NETWORK - PD (RBF NN -
PD) HYBRID***

Rizalun Iqbal
NRP 2212106024

Advisor
Dr. Ir. Mochammad Rameli
Ir. Rusdhianto Effendi A.K, M.T

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2015

**PENGATURAN KECEPATAN MOTOR INDUKSI 3 FASA
MENGUNAKAN METODE VEKTOR KONTROL BERBASIS
JARINGAN SYARAF TIRUAN *RADIAL BASIS FUNCTION* – PD
(JST RBF – PD) *HYBRID***

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui :

Dosen Pembimbing I

Dosen Pembimbing II

Dr.Ir.Mochammad Rameli
NIP.19541212 1981031002

Ir.Rusdhianto Effendie A.K.MT
NIP.19620603 198903 1 002



**SURABAYA
JANUARI, 2015**

PENGATURAN KECEPATAN MOTOR INDUKSI 3 FASA MENGGUNAKAN METODE VEKTOR KONTROL BERBASIS JARINGAN SYARAF TIRUAN *RADIAL BASIS FUNCTION* – PD (JST RBF – PD) *HYBRID*

Nama : Rizalun Iqbal
Pembimbing I : Dr. Ir. Mochammad Rameli
Pembimbing II : Ir. Rusdhianto Effendi A.K, M.T.

ABSTRAK

Pengaturan kecepatan merupakan salah satu permasalahan yang terjadi pada motor induksi. Hal ini karena sifatnya yang *nonlinear* dan terdapat parameter yang bersifat tidak pasti, salah satunya ialah beban yang berubah-ubah. Vektor kontrol merupakan salah satu metode pengontrolan yang dapat diterapkan pada motor induksi. Pada pengontrolan ini pengaturan fluks dan torsi dapat dilakukan secara terpisah. Penggunaan kontroler yang diterapkan pada metode vektor kontrol menjadi hal yang penting untuk menghasilkan kualitas kontrol yang baik. Pada tugas akhir ini Jaringan Syaraf Tiruan *Radial Basis Function* – PD (JST RBF - PD) Hybrid digunakan sebagai kontroler. Berdasarkan percobaan yang dilakukan penggunaan JST RBF – PD Hybrid sebagai kontroler kecepatan mampu mengatasi perubahan parameter beban pada motor induksi. Pada beban langsung, waktu yang dibutuhkan untuk mencapai keadaan tunak ialah rata-rata sebesar 2,68 detik saat pembebanan dan rata-rata sebesar 3,18 detik saat lepas beban dengan *undershoot* rata-rata 0,13%. Pada beban gradual, waktu yang dibutuhkan untuk mencapai keadaan tunak ialah rata-rata sebesar 2,3 detik saat beban konstan (minimal, nominal, maksimal) dan rata-rata sebesar 1,88 detik saat beban bernilai nol dengan *undershoot* rata-rata 0,06%.

Kata kunci : Kecepatan, Kontroler JST RBF – PD Hybrid, Motor Induksi, Vektor Kontrol.

SPEED CONTROL OF 3 PHASE INDUCTION MOTOR USING VECTOR CONTROL METHOD BASED RADIAL BASIS FUNCTION NEURAL NETWORK – PD (RBF NN - PD) HYBRID

Name : Rizalun Iqbal
Advisor I : Dr. Ir. Mochammad Rameli
AdvisorII : Ir. Rusdhianto Effendi A.K, M.T.

ABSTRACT

Control of speed is one of many problem that occur in induction motor. This is cause the non linear characteristic and there are some parameter are uncertain, one of them is variabel load. Vector control is a control method that can be applied to the induction motor. In this method flux and torque settings can be controled separately The use of the controller that applied to the vector control method becomes important to produce a good quality of signal control. In this final project, Radial Basis Function Neural Network - PD Hybrid (RBFNN - PD Hybrid) used as a controller. Based on experiments, RBFNN - PD Hybrid as speed controller is able to solve changing of load parameter in induction motor. In instant load, the time needed to reach steady-state is average of 2,68 seconds while loading and average of 3,18 seconds while the load is released with undershoot about 0,13 %. In gradual load , the time needed to reach steady-state is average of 2,3 seconds while constant load and average of 1,88 seconds while load is zero with undershoot about 0,06%.

Keywords : *Speed, RBFNN - PD Hybrid controller, Induction Motor, Vector Control.*

KATA PENGANTAR

Segala puji serta syukur penulis ucapkan kehadiran Allah SWT atas segala nikmat serta kasih dan sayang-Nya yang telah diberikan kepada penulis, sehingga penulis dapat menyelesaikan karya tulis ini.

Tugas Akhir yang berjudul **“Pengaturan Kecepatan Motor Induksi 3 Fasa Menggunakan Metode Vektor Kontrol Berbasis Jaringan Syaraf Tiruan *Radial Basis Function* – PD (JST RBF – PD) *Hybrid*”** ini disusun guna memenuhi persyaratan untuk menyelesaikan program Sarjana dan memperoleh gelar sarjana pada Bidang Studi Teknik Sistem Pengaturan Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis mengucapkan terima kasih kepada semua pihak yang telah membantu terselesaikannya laporan Tugas Akhir ini, terutama kepada

1. Kedua orang tua yang senantiasa mendo'akan dan mendukung penulis untuk terselesaikannya Tugas Akhir ini.
2. Bapak Dr.Ir. Mochammad Rameli dan Bapak Ir. Rusdhianto Effendie A.K., MT. selaku dosen pembimbing yang selalu membimbing dan mendukung.

Semoga laporan yang disusun ini dapat memberikan manfaat dan wawasan yang lebih luas kepada pembaca. Dalam tugas akhir ini, penulis menyadari bahwa sebagai manusia penulis tidak luput dari segala keterbatasan, oleh karena itu penulis sangat mengharapkan saran dan kritik yang membangun untuk lebih meningkatkan mutu dan penyajian karya tulis ini sebagai bekal di masa akan datang.

Surabaya, Januari 2015

Penulis

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

| | HALAMAN |
|---|--------------|
| HALAMAN JUDUL | i |
| PERNYATAAN KEASLIAN TUGAS AKHIR | v |
| HALAMAN PENGESAHAN | vii |
| ABSTRAK | ix |
| ABSTRACT | xi |
| KATA PENGANTAR | xiii |
| DAFTAR ISI | xv |
| DAFTAR GAMBAR | xix |
| DAFTAR TABEL | xxiii |
| | |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Perumusan Masalah | 2 |
| 1.3 Batasan Masalah..... | 2 |
| 1.4 Tujuan | 2 |
| 1.5 Metodologi | 2 |
| 1.6 Sistematika Penulisan..... | 3 |
| 1.7 Relevansi | 4 |
| | |
| BAB II TINJAUAN PUSTAKA | 5 |
| 2.1 Motor Induksi Tiga Fasa..... | 5 |
| 2.1.1 Konstruksi Motor Induksi Tiga Fasa | 5 |
| 2.1.2 Prinsip Kerja Motor Induksi Tiga Fasa..... | 6 |
| 2.1.3 Frekuensi dan Slip Motor..... | 7 |
| 2.2 Transformasi Vektor..... | 8 |
| 2.2.1 Transformasi Clarke | 8 |
| 2.2.2 Transformasi Park | 9 |
| 2.3 Model Matematis Motor Induksi..... | 10 |
| 2.4 Vektor Kontrol/ <i>Field Oriented Control (FOC)</i> | 13 |
| 2.4.1 Analogi Pengerak DC..... | 14 |
| 2.4.2 Prinsip Vektor Kontrol | 15 |
| 2.4.3 <i>Indirect/Feedforward Vector Control</i> | 16 |
| 2.5 <i>Hysteresis Band Current Controller PWM</i> | 19 |
| 2.6 Jaringan Syaraf Tiruan..... | 20 |

| | | |
|--|---|-----------|
| 2.6.1 | Pemodelan Sel Syaraf/Neuron..... | 20 |
| 2.6.2 | Fungsi Aktivasi..... | 21 |
| 2.6.3 | Arsitektur JST | 22 |
| 2.6.4 | Pelatihan JST..... | 24 |
| 2.6.5 | Metode Pelatihan Propagasi Balik (<i>Backpropagation</i>).. | 25 |
| 2.7 | <i>Radial Basis Function Neural Network (RBFNN)</i> | 28 |
| 2.8 | Kontroler | 31 |
| 2.8.1 | Kontroler Proporsional..... | 31 |
| 2.8.2 | Kontroler Integral | 32 |
| 2.8.3 | Kontroler Proporsional-Diferensial..... | 33 |
| 2.8.4 | Kontroler RBFNN-PD Hybrid..... | 34 |
| BAB III PERANCANGAN SISTEM | | 35 |
| 3.1 | Penentuan Parameter Motor Induksi Tiga Fasa | 36 |
| 3.2 | Pemodelan Motor Induksi Tiga Fasa | 37 |
| 3.3 | Perancangan Vektor Kontrol..... | 41 |
| 3.3.1 | Perancangan Perhitungan Kecepatan Slip (ω_{sl}) | 41 |
| 3.3.2 | Perancangan Perhitungan Arus i_{qs}^* | 41 |
| 3.3.3 | Perancangan Perhitungan Arus i_{ds}^* | 42 |
| 3.3.4 | Perancangan Unit Vektor Generator | 42 |
| 3.3.5 | Perancangan Transformasi Vektor..... | 43 |
| 3.3.6 | Perancangan <i>Hysteresis Band Current Control PWM</i> ... | 44 |
| 3.3.7 | Perancangan Inverter | 44 |
| 3.4 | Perancangan Kontroler..... | 45 |
| 3.4.1 | Perancangan Kontroler JST RBF..... | 46 |
| 3.4.2 | Algoritma Pelatihan JST RBF | 47 |
| 3.4.3 | Perancangan Kontroler JST RBF – PD Hybrid | 49 |
| 3.5 | Perhitungan Beban Nominal Motor Induksi..... | 50 |
| BAB IV HASIL SIMULASI DAN ANALISIS SISTEM | | 51 |
| 4.1 | Simulasi <i>Open Loop</i> Motor Induksi..... | 51 |
| 4.2 | Simulasi Vektor Kontrol dengan Kontroler PI | 52 |
| 4.2.1 | Pengujian Tanpa Beban | 52 |
| 4.2.2 | Pengujian dengan Beban..... | 54 |
| 4.2.2.1 | <i>Beban Secara Langsung</i> | 54 |
| 4.2.2.2 | <i>Beban Secara Gradual</i> | 56 |
| 4.3 | Simulasi Vektor Kontrol dengan Kontroler JST RBF – PD Hybrid..... | 58 |

| | | |
|-----------------------|--|-----------|
| 4.3.1 | Proses Pembelajaran..... | 58 |
| 4.3.2 | Pengujian Tanpa Beban | 62 |
| 4.3.3 | Pengujian dengan Beban..... | 62 |
| 4.3.3.1 | <i>Beban Secara Langsung</i> | 63 |
| 4.3.3.2 | <i>Beban Secara Gradual</i> | 65 |
| 4.4 | Pengamatan Nilai N_s dan N_r | 67 |
| BAB V | PENUTUP | 69 |
| 5.1 | Kesimpulan | 69 |
| 5.2 | Saran | 69 |
| DAFTAR PUSTAKA | | 71 |
| LAMPIRAN A | | 73 |
| LAMPIRAN B | | 76 |
| RIWAYAT HIDUP | | 79 |

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

| | HALAMAN |
|---|---------|
| Tabel 3.1 Parameter – parameter motor induksi..... | 36 |
| Tabel 4.1 Karakteristik respon kecepatan motor induksi menggunakan vektor kontrol dengan kontroler PI tanpa beban | 53 |
| Tabel 4.2 Nilai bobot hasil <i>learning</i> | 60 |
| Tabel 4.3 Parameter JST RBF..... | 61 |
| Tabel 4.4 Karakteristik kecepatan motor induksi saat pembebanan menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid | 64 |
| Tabel 4.5 Karakteristik kecepatan motor induksi saat pembebanan menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid | 64 |
| Tabel 4.6 Karakteristik kecepatan motor induksi saat beban konstan menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid | 66 |
| Tabel 4.7 Karakteristik kecepatan motor induksi saat beban bernilai nol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid..... | 66 |

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

| | HALAMAN |
|---|---------|
| Gambar 2.1 Bagian – bagian motor induksi..... | 5 |
| Gambar 2.2 Macam bentuk rotor..... | 6 |
| Gambar 2.3 Transformasi vektor..... | 9 |
| Gambar 2.4 Rangkaian ekivalen $d-q$ motor induksi 3 fasa..... | 10 |
| Gambar 2.5 Metode pengontrolan motor induksi..... | 13 |
| Gambar 2.6 Analogi vektor kontrol..... | 14 |
| Gambar 2.7 Prinsip vektor kontrol pada motor induksi..... | 15 |
| Gambar 2.8 Phasor diagram <i>indirect vector control</i> | 16 |
| Gambar 2.9 Diagram blok <i>indirect vector control</i> | 18 |
| Gambar 2.10 Prinsip kerja <i>hysteresis band current control PWM</i> ... | 19 |
| Gambar 2.11 Pemodelan neuron | 20 |
| Gambar 2.12 Fungsi aktivasi..... | 22 |
| Gambar 2.13 Jaringan syaraf tiruan lapisan tunggal..... | 23 |
| Gambar 2.14 Jaringan syaraf tiruan <i>multi layer</i> | 23 |
| Gambar 2.15 Jaringan syaraf tiruan <i>recurent</i> | 24 |
| Gambar 2.16 Konfigurasi RBFNN..... | 29 |
| Gambar 2.17 Kontroler RBFNN-PD Hybrid | 34 |
| Gambar 3.1 Diagram alir perancangan sistem..... | 35 |
| Gambar 3.2 Blok simulasi pemodelan motor induksi tiga fasa | 39 |
| Gambar 3.3 Sumber 3 fasa..... | 40 |
| Gambar 3.4 Transformasi $a-b-c$ ke $d-q$ | 40 |
| Gambar 3.5 Blok simulasi perhitungan kecepatan slip | 41 |
| Gambar 3.6 Blok simulasi perhitungan fluks..... | 41 |
| Gambar 3.7 Blok simulasi perhitungan arus i_{qs}^* | 42 |
| Gambar 3.8 Blok simulasi perhitungan arus i_{ds}^* | 42 |
| Gambar 3.9 Unit vektor generator | 43 |
| Gambar 3.10 Blok simulasi transformasi Park ($d-q$ ke $\alpha-\beta$)..... | 43 |
| Gambar 3.11 Blok simulasi transformasi Clarke ($\alpha-\beta$ ke $a-b-c$)..... | 44 |
| Gambar 3.12 Blok simulasi <i>hysteresis band current control PWM</i> .. | 44 |
| Gambar 3.13 Blok simulasi <i>inverter</i> | 45 |
| Gambar 3.14 Diagram blok sistem dengan kontroler JST RBF – PD Hybrid..... | 45 |
| Gambar 3.15 Struktur JST RBF | 46 |
| Gambar 3.16 Blok simulasi kontroler JST RBF-PD Hybrid | 49 |

| | | |
|--------------------|--|----|
| Gambar 4.1 | Respon kecepatan motor induksi | 52 |
| Gambar 4.2 | Respon torsi motor induksi..... | 52 |
| Gambar 4.3 | Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler PI tanpa beban..... | 53 |
| Gambar 4.4 | Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler PI tanpa beban..... | 53 |
| Gambar 4.5 | Representasi pemberian beban secara langsung..... | 54 |
| Gambar 4.6 | Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler PI dan beban langsung | 55 |
| Gambar 4.7 | Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler PI dan beban langsung..... | 56 |
| Gambar 4.8 | Representasi pemberian beban secara <i>gradual</i> | 56 |
| Gambar 4.9 | Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler PI dan beban <i>gradual</i> | 57 |
| Gambar 4.10 | Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler PI dan beban <i>gradual</i> | 57 |
| Gambar 4.11 | Respon kecepatan motor induksi hasil pemodelan dan <i>real plant</i> | 59 |
| Gambar 4.12 | Proses <i>learning</i> kontroler | 60 |
| Gambar 4.13 | Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid tanpa beban | 61 |
| Gambar 4.14 | Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid tanpa beban | 62 |
| Gambar 4.15 | Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban langsung..... | 63 |
| Gambar 4.16 | Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban langsung | 63 |
| Gambar 4.17 | Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban <i>gradual</i> | 65 |

| | | |
|--------------------|---|----|
| Gambar 4.18 | Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban <i>gradual</i> | 65 |
| Gambar 4.19 | Pengamatan nilai N_s dan N_r dengan vektor kontrol.. | 67 |
| Gambar 4.20 | Pengamatan nilai N_s dan N_r tanpa vektor kontrol | 69 |

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

1.1 Latar Belakang

Motor listrik sebagai penggerak sangat berperan penting dalam dunia industri. Jenis penggerak yang sering dipakai adalah motor AC khususnya motor induksi tiga fasa. Motor ini memiliki beberapa kelebihan dibandingkan motor jenis yang lain seperti handal dalam pengoperasian serta harga dan perawatan yang murah. Permasalahan pada motor AC ialah mengenai pengaturan kecepatan karena karakteristiknya yang bersifat *nonlinear* dan terdapat parameter yang nilainya bersifat tidak pasti seperti *temperature-dependent rotor resistance*, fluks magnetik, koefisien gesek dan beban yang berubah-ubah.

Metode pengontrolan yang biasa diterapkan pada motor induksi secara umum terdiri dari skalar kontrol dan vektor kontrol. Pada aplikasinya skalar kontrol relatif lebih mudah dalam pengontrolannya tetapi memiliki kelemahan yaitu *coupling effect* yang menyebabkan ketika satu variabel diatur maka akan berpengaruh terhadap variabel lainnya. Berbeda dengan vektor kontrol, pada metode ini motor induksi dapat dikontrol seperti motor DC eksitasi terpisah sehingga memungkinkan pengontrolan variabel secara terpisah. Variabel yang dimaksud ialah fluks dan torsi [1]. Pada setiap metode kontrol yang diterapkan, kinerja kontrol sangat bergantung pada estimasi algoritma dan akurasi dari pemodelan motor.

Sitem dinamik dari motor AC ini memiliki banyak parameter yang tidak pasti sehingga membuat metode kontrol klasik dengan atau tanpa sensor menjadi cukup sulit untuk selalu mencapai kualitas kontrol yang baik, terutama ketika beban berubah [2]. Berdasarkan hal tersebut, metode kontrol adaptif, metode identifikasi secara *online* dan jaringan syaraf tiruan merupakan metode kontrol yang mendukung untuk diterapkan.

1.2 Perumusan Masalah

Perumusan masalah yang dibahas pada tugas akhir ini ialah mengatur kecepatan motor induksi tiga fasa menggunakan metode vektor kontrol di mana Jaringan Syaraf Tiruan *Radial Basis Function – PD* (JST RBF - PD) *Hybrid* diterapkan sebagai kontroler agar dihasilkan

kualitas kontrol yang baik sehingga kondisi yang diinginkan dapat tercapai.

Permasalahan – permasalahan yang dibahas pada penelitian ini meliputi pemodelan motor induksi tiga fasa, perancangan metode vektor kontrol, perancangan kontroler Jaringan Syaraf Tiruan *Radial Basis Function* – PD (JST RBF - PD) *Hybrid* dan simulasi sistem hasil desain.

1.3 Batasan Masalah

Permasalahan pada Tugas Akhir ini dibatasi hanya sebatas simulasi dengan menggunakan simulink pada perangkat lunak MATLAB serta pemodelan yang dilakukan tidak menggunakan riil *plant* tetapi didasarkan pada rangkaian ekivalen motor induksi tiga fasa dengan menggunakan parameter-parameter yang telah ditentukan.

1.4 Tujuan

Tujuan penelitian ini adalah merancang kontroler Jaringan Syaraf Tiruan *Radial Basis Function* – PD (JST RBF - PD) *Hybrid* yang diterapkan pada metode vektor kontrol untuk mengatur kecepatan motor induksi sehingga dapat memenuhi kondisi yang diinginkan.

1.5 Metodologi

Metodologi yang digunakan dalam pengerjaan Tugas Akhir ini meliputi:

- a. Studi Literatur
Studi Literatur dilakukan untuk mendapatkan informasi mengenai sistem yang dirancang. Informasi diperoleh melalui buku teks, jurnal, artikel, media *online*, dan lain-lain.
- b. Pemodelan Sistem
Pada tahap ini metode vektor kontrol diterapkan sebagai teknik pengontrolan motor induksi yang dibuat. Dalam hal ini jenis metode vektor kontrol yang digunakan ialah *indirect or feedforward vector control*.
- c. Desain Kontroler
Pada tahap ini dilakukan perancangan pada kontroler yang digunakan yaitu dengan Jaringan Syaraf Tiruan *Radial Basis Function* – PD (JST RBF - PD) *Hybrid*.
- d. Simulasi
Pemodelan sistem dan hasil desain kontroler disimulasikan dengan menggunakan simulink pada perangkat lunak MATLAB.

- e. Penyusunan Buku Tugas Akhir
Penyusunan buku Tugas Akhir meliputi pendahuluan, teori penunjang, perancangan sistem, hasil simulasi, serta penutup.

1.6 Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dalam lima bab yang masing-masing membahas permasalahan yang berhubungan dengan Tugas Akhir yang telah dibuat dengan sistematika penulisan sebagai berikut:

Bab I : Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, metodologi penelitian, sistematika laporan, dan relevansi.

Bab II : Tinjauan Pustaka

Bab ini membahas tinjauan pustaka yang membantu penelitian, di antaranya adalah teori motor induksi, teori vektor kontrol, teori jaringan syaraf tiruan, teori *Radial Basis Function Neural Network (RBFNN)*, dan teori kontroler PID.

Bab III : Perancangan Sistem

Bab ini membahas perancangan sistem yang meliputi pemodelan motor induksi, perancangan metode vektor kontrol serta perancangan kontroler Jaringan Syaraf Tiruan *Radial Basis Function – PD (JST RBF - PD) Hybrid* pada perangkat lunak MATLAB, sehingga membantu pembaca dalam memahami sistem yang telah dirancang.

Bab IV : Simulasi dan Analisa Sistem

Bab ini memuat hasil simulasi kontroler pada sistem dan analisisnya.

Bab V : Penutup

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.7 Relevansi

Hasil dari Tugas Akhir ini diharapkan dapat berkontribusi dalam pengembangan penelitian tentang motor induksi tiga fasa serta diharapkan dapat digunakan sebagai referensi untuk implementasi ataupun pengembangan di masa mendatang.

[Halaman ini sengaja dikosongkan]

BAB II

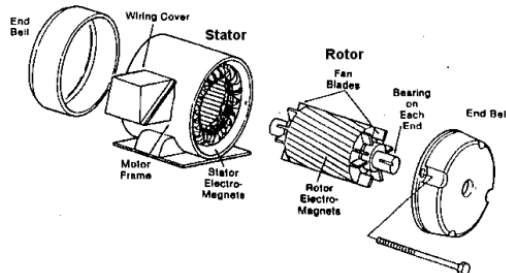
TINJAUAN PUSTAKA

2.1 Motor Induksi Tiga Fasa

Motor induksi tiga fasa merupakan motor listrik arus bolak-balik yang umumnya paling banyak digunakan dalam dunia industri. Alasan yang menjadikan motor ini banyak digunakan ialah karena beberapa keuntungan yang dimilikinya yaitu konstruksinya yang kokoh, harganya relatif murah, mudah dalam melakukan perawatan, dan dapat diproduksi dengan karakteristik yang sesuai dengan kebutuhan industri. Disamping keuntungannya terdapat pula kekurangan dari motor induksi tiga fasa seperti kecepatannya tidak mudah dikontrol dan arus *starting* besar dapat mencapai 5 sampai dengan 7 kali dari arus nominal.

2.1.1 Konstruksi Motor Induksi Tiga Fasa [3]

Konstruksi motor induksi dapat ditunjukkan seperti pada Gambar 2.1

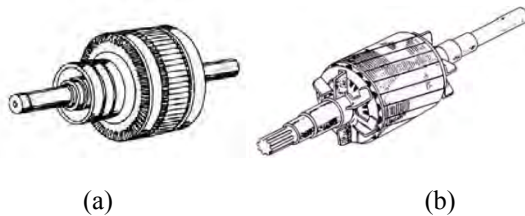


Gambar 2.1 Bagian –bagian motor induksi

Terdapat dua bagian utama pada motor induksi, yaitu bagian diam (stator) dan bagian berputar (rotor). Kedua bagian tersebut dipisahkan oleh celah udara yang sempit (air gap) dengan jarak antara 0,4 mm sampai 4 mm. Inti stator motor induksi dibuat dari besi silikon yang berlaminasi (berlembar-lembar), dengan tujuan mengurangi rugi-rugi besi (hysteresis dan arus eddy) yang dihasilkan oleh medan magnet. Pada bagian stator terdapat beberapa slot yang merupakan tempat kawat (konduktor) dari kumparan tiga fasa yang disebut kumparan stator, yang mana masing masing kumparan mendapatkan suplai arus tiga fasa. Jika

kumparan stator mendapatkan suplai arus 3 fasa, maka pada kumparan tersebut akan timbul fluks magnet putar. Oleh karena itu, adanya fluks magnet putar pada kumparan stator, mengakibatkan rotor berputar dengan kecepatan putar stator.

Motor induksi memiliki rotor yang tidak terhubung dengan sumber tegangan, rotor pada motor induksi mendapatkan power melalui induksi elektromagnetik. Tegangan AC yang terinduksi berasal dari medan magnet putar pada stator yang memotong batang konduktor pada rotor, sehingga timbul tegangan induksi pada rotor. Terdapat dua jenis rotor motor induksi yaitu rotor belitan (*wound* rotor) dan rotor sangkar (*squirrel-cage* rotor) ditunjukkan pada Gambar 2.2.



Gambar 2.2 Macam bentuk rotor
(a) Rotor belit (*wound*), (b) Rotor sangkar (*squirrel-cage*)

Jenis rotor yang akan digunakan pada tugas akhir ini adalah rotor sangkar (*squirrel cage*). Ciri khusus dari *squirrel-cage* rotor adalah ujung-ujung rotor terhubung singkat secara permanent.

2.1.2 Prinsip Kerja Motor Induksi Tiga Fasa [4]

Prinsip kerja motor induksi tiga fasa dapat dijelaskan sebagai berikut :

- a. Apabila sumber tegangan 3 fasa dipasang pada kumparan stator, maka akan timbul medan putar dengan kecepatan :

$$N_s = \frac{120f}{p} \quad (2.1)$$

di mana : f = frekuensi pada motor induksi
 p = jumlah kutub

- b. Medan putar stator akan memotong batang konduktor pada rotor, sehingga pada kumparan stator timbul gaya gerak listrik (ggl) induksi.
- c. Dikarenakan rangkaian rotor merupakan rangkaian tertutup, maka Ggl induksi yang timbul akan menyebabkan timbulnya arus (I) pada kumparan rotor.
- d. Adanya arus (I) di dalam medan magnet akan menimbulkan gaya mekanik (F) pada rotor. Jika torsi mula (T_e) yang dihasilkan oleh gaya rotor cukup besar untuk memikul torsi beban (T_L), maka rotor akan berputar searah dengan medan putar stator (N_s) dengan kecepatan putar rotor (N_r).
- e. Perbedaan antara (N_s) dan (N_r) disebut dengan slip (S), dinyatakan dengan :

$$S = \left(\frac{N_s - N_r}{N_s} \right) 100\% \quad (2.2)$$

- f. Apabila $N_s = N_r$ maka tegangan tidak akan terinduksi dan arus tidak mengalir pada kumparan jangkar rotor, sehingga tidak akan timbul torsi (T_e). Torsi akan timbul apabila N_r lebih kecil dari N_s .

2.1.3 Frekuensi dan Slip Motor

Apabila kumparan stator dihubungkan dengan sumber tegangan tiga fasa, maka timbul medan magnet yang berputar dengan kecepatan sinkron (N_s). Medan putar akan memotong penghantar atau kumparan rotor sehingga timbul gaya gerak listrik (ggl) induksi sehingga rotor berputar mengikuti medan putar stator. Bila beban bertambah, maka kopel motor akan naik sehingga arus induksi pada rotor akan membesar. Pada saat motor belum berputar, nilai frekuensi rotor sama dengan frekuensi pada belitan stator ($N_s = N_r$). Bila motor berputar, maka frekuensi rotor akan sebanding dengan perubahan slip yang terjadi pada motor tersebut [3].

Hubungan frekuensi dan slip dapat dilihat sebagai berikut [4]:

$$f_s = \frac{pN_s}{120} \quad (2.3)$$

sehingga

$$f_r = f_s \left(\frac{N_s - N_r}{N_s} \right) \quad (2.4)$$

$$f_r = f_s S \quad (2.5)$$

di mana : f_s = frekuensi stator
 f_r = frekuensi rotor

2.2 Transformasi Vektor

Motor induksi 3 fasa pada unjuk kerja dinamis merupakan sistem yang cukup kompleks karena dalam hal ini kumparan rotor berputar terhadap kumparan stator di mana setiap bagian terdiri dari 3 fasa yang variabel-variabelnya (Arus, tegangan dan fluks linkage) berubah terhadap waktu [1]. Berdasarkan hal tersebut untuk memudahkan analisa dari motor induksi 3 fasa maka sistem koordinat 3 fasa ($a-b-c$) dapat direpresentasikan menjadi sistem koordinat rotasi 2 fasa ($d-q$) atau disebut dengan transformasi vektor. Transformasi ini dilakukan dalam dua tahap yaitu :

1. Transformasi clarke yaitu mentransformasikan sistem koordinat stasioner tiga fasa ke sistem koordinat dua fasa atau disebut sistem koordinat $\alpha-\beta$.
2. Transformasi park yaitu mentransformasikan sistem koordinat $\alpha-\beta$ ke sistem koordinat $d-q$.

2.2.1 Transformasi Clarke

Transformasi sistem koordinat tiga fasa ($a-b-c$) ke dalam sistem koordinat dua fasa ($\alpha-\beta$) dapat digambarkan seperti pada Gambar 2.3 (a). Secara matematis transformasi Clarke dapat dirumuskan sebagai berikut:

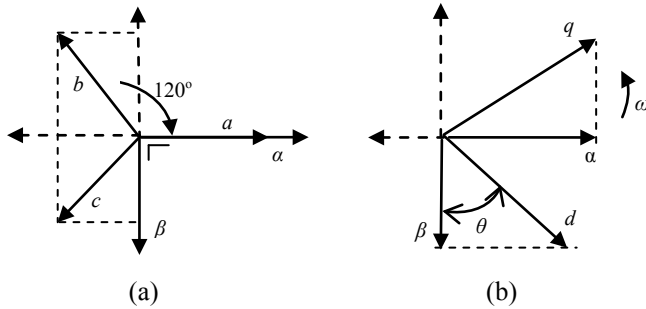
$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = k \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & -\frac{1}{2}\sqrt{3} & \frac{1}{2}\sqrt{3} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (2.6)$$

di mana : $k = 2/3$

Jika di-invers-kan menjadi :

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & -\frac{1}{2}\sqrt{3} \\ -\frac{1}{2} & \frac{1}{2}\sqrt{3} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (2.7)$$

Transformasi vektor dalam bentuk geometri dapat ditunjukkan seperti pada Gambar 2.3.



Gambar 2.3 Transformasi vektor
(a) Transformasi Clarke, (b) Transformasi Park

2.2.2 Transformasi Park

Transformasi sistem koordinat α - β ke dalam sistem koordinat rotasi d-q dapat digambarkan seperti pada Gambar 2.3 (b). Secara matematis transformasi park dapat dirumuskan sebagai berikut :

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} d \\ q \end{bmatrix} \quad (2.8)$$

Jika diinverskan menjadi

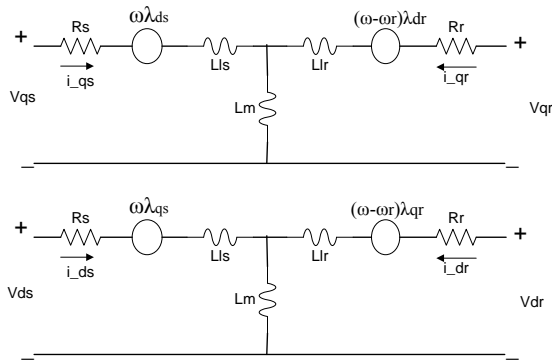
$$\begin{bmatrix} d \\ q \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (2.9)$$

2.3 Model Matematis Motor Induksi

Motor induksi identik dengan transformator, di mana kumparan primer sebagai kumparan stator dan kumparan sekunder sebagai kumparan rotor. Dengan demikian, rangkaian ekivalen motor induksi hampir sama seperti rangkaian transformator.

Dalam operasi riil ditemui permasalahan tegangan sumber yang tidak sinusoidal dan terdapat perubahan beban. Berdasarkan hal tersebut maka transformasi vektor digunakan untuk pemodelan motor induksi sehingga analisis terhadap motor induksi dapat dilakukan lebih mudah dan fleksibel. Analisis tersebut meliputi tegangan stator dan tegangan rotor, fluks stator dan fluks rotor, serta torsi elektromagnetik dari motor induksi.

Penurunan model matematis dengan menggunakan transformasi vektor dapat dibagi berdasarkan referensi frame yang digunakan yaitu tak bergerak (*stationary reference frame*), berputar (*synchronously rotating reference frame*), dan *arbitrary reference frame* [1]. Pada tugas akhir ini penurunan model matematis motor induksi dilakukan menggunakan referensi frame berputar.



Gambar 2.4 Rangkaian ekivalen d - q motor induksi 3 fasa

Berdasarkan Gambar 2.4 persamaan tegangan stator dan rotor motor induksi merupakan fungsi dari arus stator dan arus rotor, serta fungsi dari fluks yang tercakup pada kumparan. Persamaan tegangan stator dan tegangan rotor dinyatakan sebagai berikut [5] :

$$V_{qs} = r_s \cdot i_{qs} + \omega \cdot \lambda_{ds} + p \cdot \lambda_{qs} \quad (2.10)$$

$$V_{ds} = r_s \cdot i_{ds} - \omega \cdot \lambda_{qs} + p \cdot \lambda_{ds} \quad (2.11)$$

$$V_{qr} = r_r \cdot i_{qr} + (\omega - \omega_r) \lambda_{dr} + p \cdot \lambda_{qr} \quad (2.12)$$

$$V_{dr} = r_r \cdot i_{dr} - (\omega - \omega_r) \lambda_{qr} + p \cdot \lambda_{dr} \quad (2.13)$$

di mana : $p = d/dt$

Persamaan fluks pada kumparan stator dan rotor dapat dinyatakan sebagai berikut:

$$\lambda_{qs} = L_{ls} \cdot i_{qs} + L_m (i_{qs} + i_{qr}) \quad (2.14)$$

$$\lambda_{ds} = L_{ls} \cdot i_{ds} + L_m (i_{ds} + i_{dr}) \quad (2.15)$$

$$\lambda_{qr} = L_{lr} \cdot i_{qr} + L_m (i_{qs} + i_{qr}) \quad (2.16)$$

$$\lambda_{dr} = L_{lr} \cdot i_{dr} + L_m (i_{ds} + i_{dr}) \quad (2.17)$$

Persamaan tegangan dan fluks dapat dinyatakan dalam bentuk matriks seperti pada persamaan 2.18

$$\begin{bmatrix} v_{qs} \\ v_{ds} \\ v_{qr} \\ v_{dr} \end{bmatrix} = \begin{bmatrix} R_s + pL_s & \omega L_s & pL_m & \omega L_m \\ -\omega L_s & R_s + pL_s & -\omega L_m & pL_m \\ pL_m & (\omega - \omega_r)L_m & R_r + pL_r & (\omega - \omega_r)L_r \\ -(\omega - \omega_r)L_m & pL_m & -(\omega - \omega_r)L_r & R_r + pL_r \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix} \quad (2.18)$$

dengan :

$$\begin{aligned} L_s &= L_{ls} + L_m \\ L_r &= L_{lr} + L_m \end{aligned} \quad (2.19)$$

di mana : v_{qs}, v_{ds} = tegangan stator pada sumbu d-q (Volt)
 v_{qr}, v_{dr} = tegangan rotor pada sumbu d-q (Volt)
 i_{qs}, i_{ds} = arus stator pada sumbu d-q (Ampere)
 i_{qr}, i_{dr} = arus rotor pada sumbu d-q (Ampere)
 $\lambda_{qs}, \lambda_{ds}$ = fluks stator pada sumbu d-q (Webber)
 $\lambda_{qr}, \lambda_{dr}$ = fluks rotor pada sumbu d-q (Webber)

| | |
|------------|--|
| R_s | = tahanan stator (Ohm) |
| R_r | = tahanan rotor (Ohm) |
| L_{ls} | = induktansi diri stator (Henry) |
| L_{lr} | = induktansi diri rotor (Henry) |
| L_m | = induktansi mutual (Henry) |
| ω_r | = kecepatan putar elektrik rotor (rad/detik) |
| ω | = sumber frekuensi stator (rad/detik) |

Persamaan torsi yang dihasilkan oleh motor dapat dinyatakan dalam persamaan 2.20 dan 2.21 :

$$T_e = \frac{3}{2} \frac{p}{2} L_m (i_{dr} i_{qs} - i_{qr} i_{ds}) \quad (2.20)$$

$$T_e = \frac{3}{2} \frac{p}{2} \frac{L_m}{L_r} (\lambda_{dr} i_{qs} - \lambda_{qr} i_{ds}) \quad (2.21)$$

Kecepatan putaran motor induksi merupakan fungsi dari torsi elektromagnetik dan torsi beban. Perumusan untuk mendapatkan kecepatan motor induksi adalah sebagai berikut :

$$J \frac{d}{dt} \omega_m + B \omega_m = J \frac{2}{p} \frac{d}{dt} \omega_r + B \omega_r = T_e - T_L \quad (2.22)$$

$$\frac{d}{dt} \theta_r = \omega_r \quad (2.23)$$

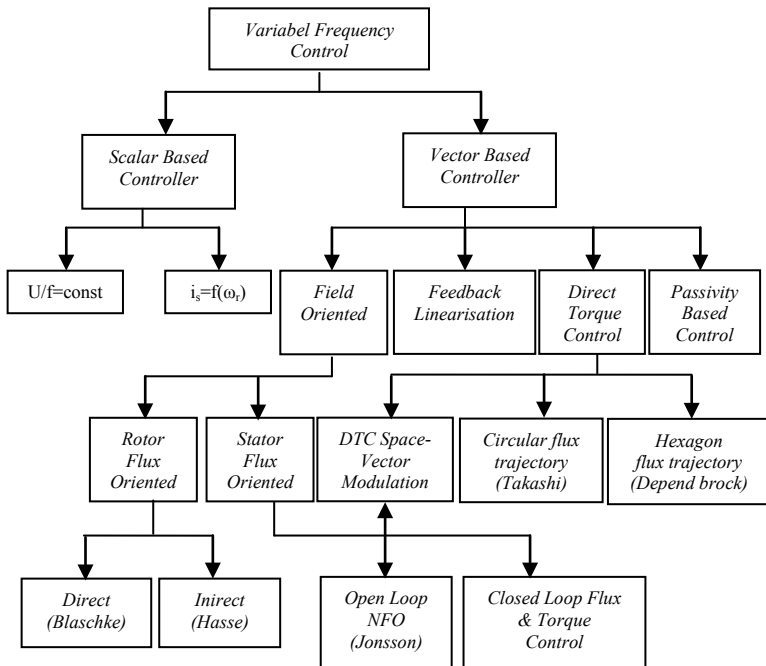
di mana : T_e = torsi elektromagnetik (N.m)
 J = momen inersia (kg.m²)
 ω_m = kecepatan putar mekanik rotor (rad/detik)
 θ_r = posisi sudut (rad)

2.4 Vektor Kontrol/ *Field Oriented Control (FOC)*

Terdapat banyak macam metode pengontrolan motor induksi, salah satunya ialah *Field Oriented Control (FOC)* atau biasa disebut vektor kontrol. Vektor kontrol adalah suatu metode pengaturan medan pada motor AC, di mana dari sistem *coupled* dirubah menjadi sistem *decoupled*. Dengan sistem ini arus penguatan dan arus beban motor

dapat dikontrol secara terpisah, dengan demikian torsi dan fluks juga dapat diatur secara terpisah, seperti halnya motor DC [1].

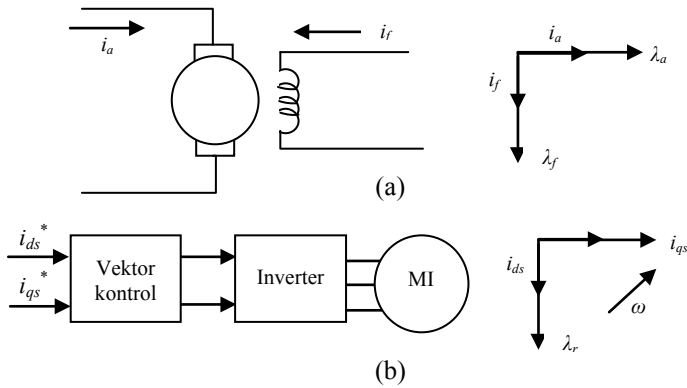
Field oriented control ini dibagi menjadi 2 jenis *rotor flux oriented* dan *stator flux oriented*. *Rotor oriented flux* juga dibagi lagi menjadi 2 jenis *direct* dan *indirect*. Pada tugas akhir metode pengontrolan yang digunakan ialah jenis *indirect / feedforward vector control*. Untuk klasifikasi metode pengontrolan motor induksi ini ditunjukkan pada Gambar 2.5 [6].



Gambar 2.5 Metode pengontrolan motor induksi

2.4.1 Analogi Pengerak DC [1]

Secara ideal pengoprasian motor induksi dengan menggunakan vektor kontrol sama seperti motor DC eksitasi terpisah. Gambar 2.6 menunjukkan analogi ini.



Gambar 2.6 Analogi vektor kontrol

(a) Motor DC eksitasi terpisah, (b) Motor induksi dengan vektor kontrol

Pada motor DC eksitasi terpisah dengan mengabaikan efek reaksi jangkar dan saturasi medan maka torsi yang dibangkitkan:

$$T_e = K_t' i_a i_f = K_t \lambda_a \lambda_f \quad (2.24)$$

di mana: i_a = arus jangkar/rotor (Ampere)

i_f = arus medan (Ampere)

λ_a = fluks jangkar (Weber)

λ_f = fluks medan (Weber)

Pada motor DC eksitasi terpisah ketika torsi (T_e) diatur dengan mengatur i_a maka λ_f tidak akan terpengaruh. Begitu pula ketika i_f diatur maka yang berpengaruh hanyalah nilai λ_f , sehingga menyebabkan *coupling effect* tidak terjadi pada motor DC eksitasi terpisah.

Berdasarkan hal tersebut maka dengan memodelkan motor induksi pada referensi frame berputar ($d-q$ model) yang diatur dengan menggunakan vektor kontrol dapat dianalogikan $i_{ds} = i_f$ dan $i_{qs} = i_a$. Torsi yang dibangkitkan dapat dirumuskan sebagai berikut:

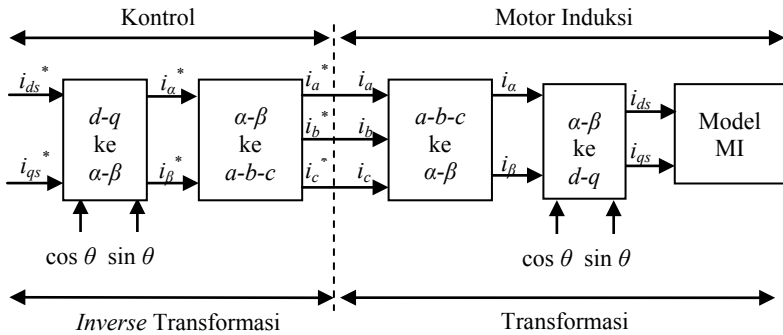
$$T_e = K_t' i_{qs} i_{ds} = K_t \hat{\lambda}_r i_{qs} \quad (2.25)$$

di mana : i_{qs} , i_{ds} = arus masukan pada stator
 λ_r = fluks pada rotor

Berdasarkan Gambar 2.6 (b), i_{qs}^* adalah arus kontrol untuk i_{qs} dan i_{ds}^* adalah arus kontrol untuk i_{ds} yang diberikan pada motor induksi. Oleh karena itu ketika torsi akan diatur dengan menggunakan i_{qs}^* maka tidak akan berpengaruh pada λ_r dan ketika i_{ds}^* diatur hanya akan berpengaruh pada λ_r . Berdasarkan hal tersebut maka pengaturan fluks dan torsi dapat dilakukan secara terpisah (*decoupling*) dengan mengatur i_{qs}^* dan i_{ds}^* .

2.4.2 Prinsip Vektor Kontrol [1]

Prinsip vektor kontrol untuk diterapkan pada motor induksi dapat dijelaskan berdasarkan Gambar 2.7.



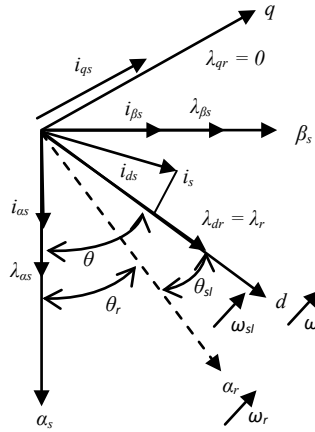
Gambar 2.7 Prinsip vektor kontrol pada motor induksi

Pada gambar 2.7 pemodelan motor induksi disajikan dalam d-q model (referensi frame berputar). Pada bagian kontrol nilai i_{qs}^* dan i_{ds}^* diatur untuk mengatur masukan aktual i_{qs} dan i_{ds} pada motor induksi. Antara bagian kontrol dan motor induksi dalam hal ini nilai arus yang diinginkan (i_a^* , i_b^* , i_c^*) dirubah menjadi masukan motor induksi (i_a , i_b , i_c) dengan menggunakan inverter. Pada prosesnya perubahan frame yaitu dari koordinat $\alpha\beta$ menjadi d-q dan sebaliknya digunakan komponen $\cos \theta$ dan $\sin \theta$. Komponen ini disebut dengan unit vektor.

2.4.3 Indirect / Feedforward Vector Control [1]

Secara umum terdapat dua jenis metode vektor kontrol dengan menggunakan rotor flux oriented yaitu *direct/feedback vector control* dan *indirect/feedforward vector control*. Perbedaan keduanya terletak pada bagaimana unit vektor dihasilkan.

Prinsip dari *indirect vector control* ini dapat dijelaskan dengan menggunakan fasor diagram seperti tampak pada Gambar 2.8.



Gambar 2.8 Phasor diagram *indirect vector control*

Berdasarkan phasor diagram tersebut untuk melakukan *decoupling control*, fluks (λ_r) pada arus i_{ds} harus segaris dengan sumbu d dan komponen torsi (T_e) pada arus i_{qs} harus segaris dengan sumbu q.

Penurunan persamaan untuk *indirect vector control* ini dapat dilakukan dengan menggunakan rangkaian ekivalen d-q (Gambar 2.4). Dengan menggunakan persamaan fluks rotor (persamaan 2.16 dan 2.17) maka persamaan dapat dibentuk menjadi :

$$i_{dr} = \frac{1}{L_r} \lambda_{dr} - \frac{L_m}{L_r} i_{ds} \quad (2.26)$$

$$i_{qr} = \frac{1}{L_r} \lambda_{qr} - \frac{L_m}{L_r} i_{qs} \quad (2.27)$$

Berdasarkan gambar 2.8, maka untuk pengaturan secara terpisah (*decoupling control*) diinginkan:

$$\lambda_{qr} = 0 \quad (2.28)$$

Selanjutnya dengan mensubstitusi persamaan 2.26 dan 2.27 ke persamaan tegangan rotor (persamaan 2.12 dan 2.13) :

$$\hat{\lambda}_r = \frac{L_m}{T_r s + 1} \quad (2.29)$$

Jika fluks $\hat{\lambda}_r$ konstan maka :

$$\hat{\lambda}_r = L_m i_{ds} \quad (2.30)$$

Kecepatan slip (ω_{sl}) didapatkan dari perhitungan pada persamaan 2.31.

$$\omega_{sl} = \frac{L_m i_{qs}}{\hat{\lambda}_r T_r} \quad (2.31)$$

di mana: $T_r = L_r / R_r$

$$s = d/dt$$

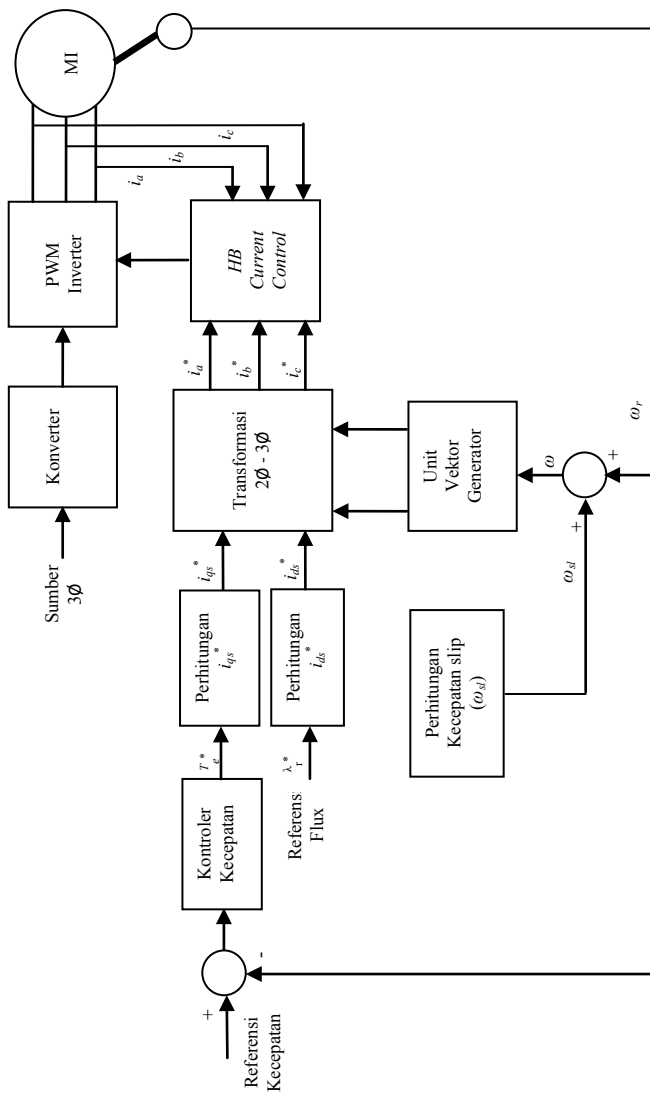
$$\hat{\lambda}_r = \lambda_{dr}$$

$$\omega_{sl} = \omega - \omega_r$$

Perhitungan torsi dapat dihitung berdasarkan persamaan 2.21, sehingga didapatkan:

$$T_e = \frac{3}{2} \frac{p}{2} \frac{L_m}{L_r} \left(\hat{\lambda}_r i_{qs} \right) \quad (2.32)$$

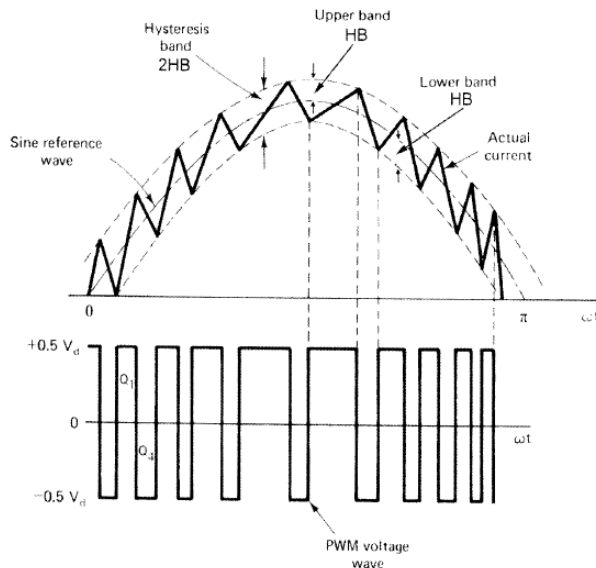
Skema pengontrolan motor induksi dengan menggunakan *indirect vector control* untuk pengaturan kecepatan dengan fluks referensi konstan dapat digambarkan dalam bentuk diagram blok seperti pada Gambar 2.9.



Gambar 2.9 Digram blok *indirect vector control*

2.5 Hysteresis Band Current Controller PWM [1]

Hysteresis band current control PWM adalah metode pembangkit sinyal PWM yang pengontrolnya didasarkan pada arus umpan balik (arus aktual) yang dibandingkan terhadap arus referensi di mana arus aktual secara terus menerus melakukan *tracking* terhadap arus *command* dengan nilai hysteresis band tertentu. Prinsip kerja dari *hysteresis band current control PWM* ini dapat ditunjukkan seperti pada Gambar 2.10.



Gambar 2.10 Prinsip kerja *hysteresis band current control PWM*

Berdasarkan Gambar 2.10 dapat dilihat bahwa sinyal PWM dihasilkan dari selisih antara arus aktual dan arus referensi. Ketika selisih bernilai positif maka sinyal yang dihasilkan ialah *high* (1) dan ketika selisih bernilai negatif maka sinyal yang dihasilkan ialah *low* (0). *Hysteresis band current control PWM* ini berpengaruh terhadap *switching* frekuensi yang dihasilkan oleh inverter.

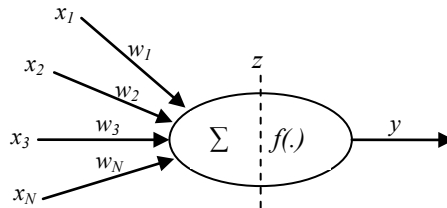
2.6 Jaringan Syaraf Tiruan

Jaringan syaraf tiruan dapat didefinisikan sebagai salah satu bentuk penyederhanaan dari jaringan syaraf biologis yang dinyatakan dalam formulasi matematika jaringan simbol. Komponen penyusun jaringan syaraf tiruan ini terdiri dari beberapa neuron, di mana neuron-neuron tersebut memiliki hubungan antara satu dengan lainnya. Hubungan antara masukan dengan didefinisikan melalui suatu fungsi tertentu.

Jaringan syaraf tiruan ini dapat dibentuk berdasarkan struktur pembentuknya atau disebut arsitektur JST. Pada prosesnya terjadi proses pembelajaran/pelatihan agar tujuan/hasil sesuai dengan nilai yang diinginkan. Metode pelatihan yang digunakan dapat bermacam-macam sesuai dengan kebutuhannya.

2.6.1 Pemodelan Sel Syaraf /Neuron

Jaringan syaraf tiruan terdiri dari beberapa neuron, dan memiliki hubungan antara neuron-neuron tersebut. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima melalui sambungan keluarnya menuju ke neuron-neuron yang lain. Pada jaringan syaraf, hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut. Neuron /sel syaraf dapat dimodelkan seperti pada Gambar 2.11.



Gambar 2.11 Pemodelan neuron

di mana : x = *input*

w = bobot

Σ = proses pengumpulan informasi

$f(.)$ = proses penarikan kesimpulan

y = *output*

Proses pengumpulan informasi didefinisikan sebagai jumlah dari perkalian nilai *input* (x) dengan bobot (w) sehingga dapat dituliskan sebagai berikut :

$$z = \sum_{i=1}^N w_i x_i \quad (2.33)$$

Proses penarikan kesimpulan didefinisikan sebagai fungsi aktivasi dari proses sebelumnya sehingga dapat dituliskan sebagai berikut:

$$y = f(z) \quad (2.34)$$

2.6.2 Fungsi Aktivasi

Fungsi aktivasi merupakan fungsi pengolah jumlahan data *input* menjadi data *output*. Beberapa fungsi aktivasi yang biasa digunakan diantaranya :

a. *Linear*

Fungsi linear ini dapat didefinisikan sebagai berikut :

$$y = \lambda z \quad (2.35)$$

di mana : $\lambda = \text{learning rate}$

b. *Unipolar Sigmoid*

Fungsi *unipolar sigmoid* ini dapat didefinisikan sebagai berikut:

$$y = \frac{1}{1 + e^{-\lambda z}} \quad (2.36)$$

c. *Bipolar Sigmoid*

Fungsi *bipolar sigmoid* ini dapat didefinisikan sebagai berikut :

$$y = \tanh(\lambda z) \quad (2.37)$$

d. Gaussian

Fungsi gaussian ini dapat didefinisikan sebagai berikut :

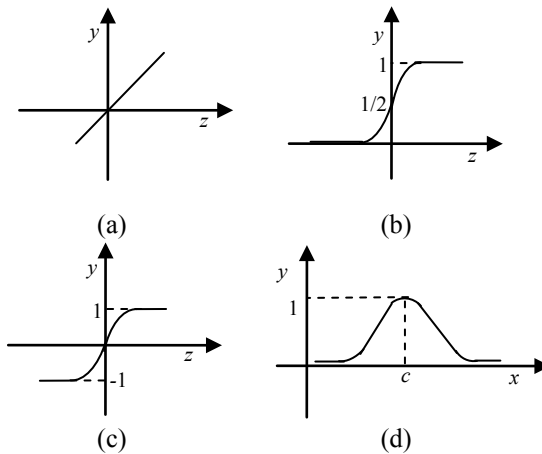
$$y = e^{-\left(\frac{x-c}{\sigma}\right)^2} \quad (2.38)$$

di mana: x = input

c = nilai tengah

σ = lebar gaussian

representasi setiap fungsi tersebut dapat digambarkan dalam bentuk kurva seperti pada Gambar 2.12.



Gambar 2.12 Fungsi aktivasi

(a) *Linear* , (b) *Unipolar Sigmoid*, (c) *Bipolar Sigmoid*, (d) *Gaussian*

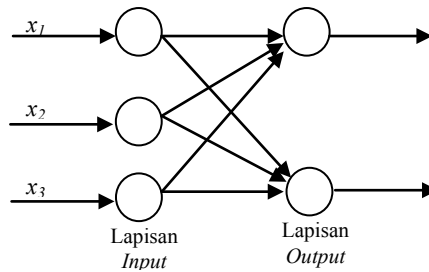
2.6.3 Arsitektur JST

Arsitektur jaringan syaraf tiruan merupakan pengaturan neuron dalam *layer* dan hubungan-hubungannya. Arsitektur sebuah jaringan akan menentukan keberhasilan target yang akan dicapai, karena tidak

semua permasalahan dapat diselesaikan dengan arsitektur yang sama. Secara umum arsitektur JST dapat dibagi menjadi beberapa yaitu :

a. Lapisan Tunggal/ Single Layer

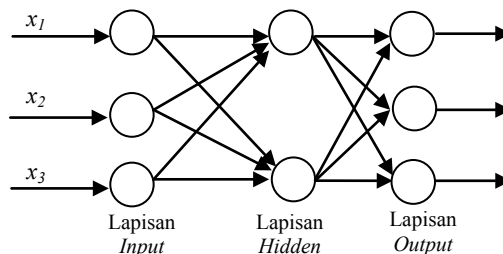
Pada jaringan lapisan tunggal, neuron-neuron dapat dikelompokkan menjadi dua bagian, yaitu unit-unit *input* dan unit-unit *output*. Unit-unit *input* menerima masukan dari luar sedangkan unit-unit *output* akan mengeluarkan respon dari jaringan sesuai dengan masukannya. Jaringan lapisan tunggal ini dapat ditunjukkan seperti pada Gambar 2.13.



Gambar 2.13 Jaringan syaraf tiruan lapisan tunggal

b. Banyak Lapisan/ Multi Layer

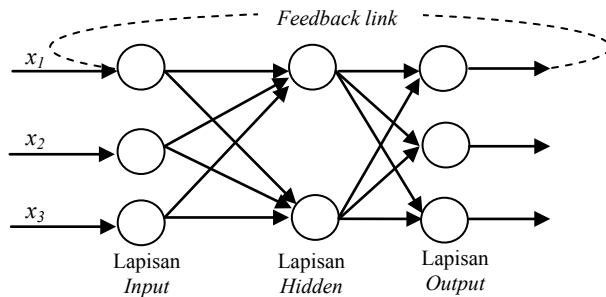
Susunan jaringan *multilayer*, selain terdiri dari unit-unit *input* dan unit-unit *output*, juga terdapat unit-unit tersembunyi (*hidden layer*). Jumlah unit *hidden* tergantung pada kebutuhan. Semakin kompleks jaringan, unit *hidden* yang dibutuhkan makin banyak, demikian pula jumlah *layer*nya. Jaringan *multi layer* ini dapat ditunjukkan seperti pada Gambar 2.14.



Gambar 2.14 Jaringan syaraf tiruan *multi layer*

c. *Recurrent*

Arsitektur ini mempunyai alur untuk mengembalikan nilai *output* untuk memberikan sinyal ke *layer input* (*feedback link*). Model ini setidaknya harus memiliki minimal satu *feedback link*. Recurrent Networks bisa ditingkatkan akurasi dengan menambahkan *hidden layer*. Jaringan *recurrent* ini dapat ditunjukkan seperti pada Gambar 2.15.



Gambar 2.15 Jaringan syaraf tiruan *recurrent*

Selain arsitektur yang dijelaskan masih banyak jenis-jenis arsitektur yang dapat digunakan untuk membentuk jaringan syaraf tiruan, salah satunya ialah jaringan syaraf tiruan yang dibentuk berdasarkan struktur fungsi basis radial (*radial basis function*).

2.6.4 Pelatihan JST [5]

Pada JST, belajar adalah proses pembentukan konfigurasi harga-harga bobot dari jaringan. Pembentukan ini memiliki tujuan akhir agar *input-input* yang diberikan padanya akan direspon melalui bobot-bobot tersebut, sehingga menghasilkan *output* yang sesuai dengan target atau mendekati untuk *input* yang bersangkutan. Terdapat banyak cara di dalam proses pembelajaran dari JST, secara umum dapat dikategorikan dalam 2 jenis proses, yaitu :

a. *Supervised training*

Pada tipe belajar dengan pengawasan, tiap pola *input* memiliki pola target. Sehingga pada pembelajaran tipe ini, masing-masing *input* memiliki pasangan *output* yang

bersesuaian. Pada proses belajarnya, bobot-bobot dibangun menuju kesesuaian respon pasangan *input-output* dari pola yang diajarkan, yaitu tiap diberikan pola *input* yang diajarkan, pembobotnya dapat memberikan respon dan menghasilkan pola *output* yang sesuai dengan target dari pola *input* tersebut. Dalam hal ini, dapat diterapkan toleransi kesalahan *output* respon terhadap target yang seharusnya. *Error* digunakan untuk mengubah bobot sambungan sehingga kesalahan akan semakin kecil dalam siklus pelatihan berikutnya. Pada tugas akhir ini akan digunakan tipe pembelajaran di atas.

b. *Unsupervised training*

Algoritma ini tidak membutuhkan vektor target untuk keluarannya, sehingga tidak ada perbandingan untuk menentukan respon yang ideal. Kumpulan pola pelatihan hanya terdiri dari vektor masukan dan, algoritma pelatihan berfungsi sebagai pengubah/update pembobot (*weight*) jaringan untuk menghasilkan pola vektor, sehingga penerapan 2 vektor pelatihan suatu vektor lain yang cukup sejenis menghasilkan pola keluaran yang sama.

Dalam proses pelatihan, jaringan mengklasifikasikan pola-pola masukan menjadi kelompok yang sejenis. Penerapan suatu vektor dari suatu kelas tertentu pada masukannya akan menghasilkan vektor keluaran yang spesifik, namun tak ada cara untuk menentukan terlebih dahulu pada pelatihannya, yang akan menghasilkan pola keluaran tertentu dengan satu vektor masukan dari kelas tertentu.

2.6.5 Metode Pelatihan Propagasi Balik (*Backpropagation*)

Metode pelatihan propagasi balik merupakan suatu algoritma yang digunakan untuk menentukan nilai bobot agar target atau nilai keluaran yang diinginkan tercapai.

Secara garis besar, mengapa algoritma ini disebut sebagai propagasi balik, dapat dideskripsikan sebagai berikut: Ketika Jaringan diberikan pola *input* sebagai pola pelatihan maka pola tersebut menuju ke unit-unit pada lapisan *hidden*/tersembunyi untuk diteruskan ke unit-unit lapisan *output*. Kemudian unit-unit lapisan *output* memberikan tanggapan yang disebut sebagai *output* jaringan. Saat *output* jaringan tidak sama dengan *output* yang diharapkan maka *output* akan menyebar

mundur (*backward*) pada lapisan tersembunyi diteruskan ke unit pada lapisan *input*. Oleh karenanya maka mekanisme pelatihan tersebut dinamakan *backpropagation*/propagasi balik.

Prinsip algoritma *backpropagation* memiliki 3 fase, yaitu:

- a. Fase *feedforward* pada pola *input* pembelajaran.
- b. Fase kalkulasi dan *backpropagation error* yang didapat.
- c. Fase penyesuaian bobot.

Algoritma dari *backpropagation* dapat dijelaskan sebagai berikut:

- 1) Langkah 0 : Inisialisasi bobot. (sebaiknya diatur pada nilai acak yang kecil),
- 2) Langkah 1 : Jika kondisi tidak tercapai, lakukan langkah 2-9,
- 3) Langkah 2 : Untuk setiap pasangan pelatihan, lakukan langkah 3-8,

a. Fase *feed forward* :

- 4) Langkah 3 : Setiap unit *input* (x_i , $i = 1, \dots, n$) menerima sinyal x_i dan menghantarkan sinyal ini ke semua unit lapisan di atasnya (unit *hidden*),
- 5) Langkah 4 : Setiap unit *hidden* (z_j , $j = 1, \dots, p$) menjumlahkan sinyal-sinyal *input* pembobotnya,

$$z_in_j = v_{oj} + \sum_{i=1}^n x_i v_{ij} \quad (2.39)$$

di mana v_{oj} = bias pada unit *hidden* dan kemudian dihitung sinyal keluaranya sesuai dengan fungsi aktivasi yang digunakan,

$$z_j = f(z_in_j) \quad (2.40)$$

Kemudian kirimkan sinyal ini keseluruhan unit pada lapisan di atasnya (unit *output*).

- 6) Langkah 5 : Setiap unit *output* (y_k , $k = 1, \dots, m$) menjumlahkan sinyal-sinyal *input* pembobotnya,

$$y_in_k = w_{ok} + \sum_{j=1}^p z_j w_{jk} \quad (2.41)$$

di mana w_{ok} = bias pada unit *output* dan kemudian dihitung sinyal keluaranya sesuai dengan fungsi aktivasi yang digunakan,

$$y_k = f(y_{in_k}) \quad (2.42)$$

b. Fase kalkulasi dan *backpropagation error* :

- 7) Langkah 6 : Setiap unit *output* ($y_k, k = 1, \dots, m$) menerima pola target yang saling berhubungan pada masukan pola pelatihan dan hitung kesalahan informasinya,

$$\delta_k = (t_k - y_k) f'(y_{in_k}) \quad (2.43)$$

hitung koreksi bobotnya (digunakan untuk memperbaharui w_{jk} nantinya),

$$\Delta w_{jk} = \alpha \delta_k z_j \quad (2.44)$$

hitung koreksi biasnya (digunakan untuk memperbaharui w_{ok} nantinya),

$$\Delta w_{ok} = \alpha \delta_k \quad (2.45)$$

dan kirimkan δ_k ke unit-unit pada lapisan dibawahnya,

- 8) Langkah 7 : Setiap unit lapisan *hidden* ($z_j, j = 1, \dots, p$) menjumlahkan hasil perubahan masukannya (dari unit-unit lapisan diatasnya),

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \quad (2.46)$$

kalikan dengan turunan fungsi aktivasinya untuk menghitung informasi kesalahannya,

$$\delta_j = \delta_{in_j} f'(z_{in_j}) \quad (2.47)$$

hitung koreksi bobotnya (digunakan untuk memperbaharui v_{ij} nanti),

$$\Delta v_{ij} = \alpha \delta_j x_i \quad (2.48)$$

dan hitung koreksi biasnya (digunakan untuk memperbaharui v_{oj} nanti),

$$\Delta v_{oj} = \alpha \delta_j \quad (2.49)$$

c. Fase peyesuaian bobot :

- 9) Langkah 8 : Setiap unit *output* ($y_k, k = 1, \dots, m$) diperbaharui bias dan bobotnya ($j = 0, \dots, p$) :

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (2.50)$$

setiap unit lapisan *hidden* ($z_j, j = 1, \dots, p$) diperbaharui bias dan bobotnya ($i = 0, \dots, n$):

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta w_{ij} \quad (2.51)$$

- 10) Uji kondisi

2.7 Radial Basis Function Neural Network (RBFNN)

Radial basis function merupakan himpunan fungsi-fungsi yang digunakan untuk merepresentasikan suatu regresi/pendekatan yang melalui sejumlah titik dengan banyak titik belok. Nilai yang dihasilkan dari fungsi basis radial ini tergantung dari nilai norm jarak yang dihasilkan atau dapat ditulis seperti pada persamaan 2.52.

$$r = \|x - \bar{x}\| \quad (2.52)$$

Fungsi basis radial ini dapat dibentuk oleh beberapa fungsi diantaranya:

- Fungsi Sigmoid
- Fungsi-fungsi trigonometri
- Fungsi gaussian
- Fungsi-fungsi kurva lonceng
- Fungsi-fungsi kurva s dan z

Radial Basis Function Neural Network (RBFNN) adalah suatu *radial basis function* yang dinyatakan dalam bentuk *neural network*/jaringan syaraf tiruan. Pada RBFNN pemrosesan sinyal dari *input layer* ke *hidden layer* sifatnya nonlinier, sedangkan dari *hidden layer* ke *output layer* bersifat linier.

Pada tugas akhir ini fungsi yang digunakan untuk membentuk RBFNN ialah fungsi gaussian. Secara umum RBFNN dengan fungsi gaussian dinyatakan dalam bentuk persamaan sebagai berikut :

$$f(x) = \sum_{i=1}^N a_i e^{-\left(\frac{x_i - \bar{x}_i}{\sigma_i}\right)^2} \quad (2.53)$$

di mana :

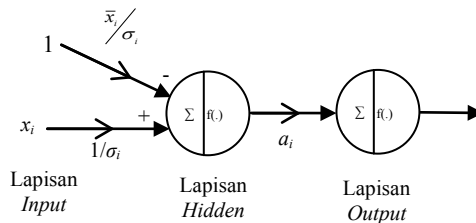
N = jumlah *input*

x_i = nilai *input*

σ_i = lebar gaussian

\bar{x}_i = nilai tengah gaussian

Dalam bentuk jaringan syaraf tiruan dengan struktur RBF dengan fungsi gaussian dapat dibentuk dari beberapa konfigurasi, salah satunya ditunjukkan pada Gambar 2.16.



Gambar 2.16 Konfigurasi RBFNN

a. Lapisan Input

Input berupa nilai x_i dan faktor pengali nilai tengah dalam hal ini diberi nilai = 1.

b. Lapisan Hidden

Pada lapisan ini nilai masukan di proses sesuai ketentuan jaringan syaraf tiruan. Pada variabel antara (z_{hl}) didapatkan sebagai berikut :

$$z_h = \left(\frac{\bar{x}_i}{\sigma_i}\right)(1) - \left(\frac{1}{\sigma_i}\right)\bar{x}_i \quad (2.54)$$

di mana : $1/\sigma_i$ = bobot *input* x_i

$$\frac{\bar{x}_i}{\sigma_i} = \text{bobot input } 1$$

Untuk hubungan *input* dan *output* digunakan fungsi aktivasi gaussian sehingga didapatkan :

$$y_h = e^{-(z_h)^2} \quad (2.55)$$

c. Lapisan Output

Pada variabel antara (z_o) didapatkan sebagai berikut :

$$z_o = a_i y_h \quad (2.56)$$

di mana : a_i = bobot y_h

Untuk hubungan *input* dan *output* digunakan fungsi aktivasi linear sehingga didapatkan :

$$y_o = \lambda z_o \quad (2.57)$$

dengan $\lambda = 1$

2.8 Kontroler

Kontroler merupakan salah satu komponen sistem pengaturan yang berfungsi mengolah sinyal umpan balik dan sinyal masukan acuan (*setpoint*) atau sinyal *error* mejadi sinyal kontrol. Sinyal *error* disini adalah selisih antara sinyal umpan balik yang dapat berupa sinyal keluaran plant sebenarnya atau sinyal keluaran terukur dengan sinyal masukan acuan [7].

2.8.1 Kontroler Proporsional [3]

Untuk kontroler proporsional, sinyal kesalahan $e(t)$ merupakan masukan kontroler sedangkan keluaran kontroler adalah sinyal kontrol $u(t)$. Secara lebih sederhana dapat dikatakan bahwa keluaran kontroler proporsional merupakan perkalian antara konstanta proporsional dengan masukannya. Hubungan antara masukan kontroler $e(t)$ dan keluaran kontroler $u(t)$ seperti pada persamaan 2.58

$$u(t) = K_p \cdot e(t) \quad (2.58)$$

di mana : K_p = penguatan proporsional

Sehingga fungsi alih kontroler proporsional dalam bentuk laplace adalah

$$\frac{U(s)}{E(s)} = K_p \quad (2.59)$$

Apapun mekanismenya dan apapun bentuk gaya operasinya, kontroler proporsional pada dasarnya merupakan suatu penguat dengan penguatan yang dapat diatur.

Kontroler proporsional memiliki dua parameter, pita proporsional (proporsional *band*) dan konstanta proporsional. Daerah kerja kontroler efektif dicerminkan oleh pita proporsional, sedangkan konstanta proporsional menunjukkan nilai faktor penguat terhadap sinyal kesalahan (K_p).

Ciri-ciri kontroler proporsional harus diperhatikan ketika kontroler tersebut diterapkan pada suatu sistem. Secara eksperimen, penggunaan kontroler proporsional harus memperhatikan ketentuan-ketentuan berikut :

- a. Jika nilai K_p kecil, kontroler proporsional hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat.
- b. Jika nilai K_p dinaikkan, respon sistem menunjukkan semakin cepat mencapai keadaan yang diinginkan. Akan tetapi jika nilai dari K_p diperbesar hingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil.

2.8.2 Kontroler Integral [3]

Untuk kontroler integral, sinyal kesalahan $e(t)$ merupakan masukan kontroler sedangkan keluaran kontroler adalah sinyal kontrol $u(t)$. Pada kontroler integral nilai masukan kontroler $u(t)$ diubah pada laju proporsional dari sinyal kesalahan $e(t)$ sehingga hubungan antara masukan kontroler $e(t)$ dan keluaran kontroler $u(t)$ ditunjukkan pada persamaan 2.60

$$u(t) = K_i \int_0^t e(t) dt \quad (2.60)$$

Dalam bentuk laplace persamaan 2.61 menjadi :

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \quad (2.61)$$

di mana K_i adalah konstanta integral yang nilainya dapat diubah – ubah.

Kontroler integral berfungsi menghasilkan respon sistem yang memiliki kesalahan keadaan mendekati nol. Jika sebuah *plant* tidak memiliki unsur integrator ($1/s$), kontroler proporsional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan nol. Keluaran kontroler ini merupakan jumlahan yang terus menerus dari perubahan masukannya. Jika sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Sinyal keluaran kontroler integral merupakan luas bidang - bidang yang dibentuk oleh kurva kesalahan penggerak. Sinyal keluaran akan berharga sama dengan harga sebelumnya ketika sinyal kesalahan berharga nol.

Ketika menggunakan kontroler integral mempunyai beberapa karakteristik berikut ini :

- a. Keluaran kontroler membutuhkan selang waktu tertentu, sehingga kontroler integral cenderung memperlambat respon.
- b. Ketika sinyal kesalahan berharga nol, keluaran kontroler akan bertahan pada nilai sebelumnya.
- c. Jika sinyal kesalahan tidak berharga nol, keluaran akan menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai K_i .

2.8.3 Kontroler Proporsional-Diferensial [7]

Kontroler proporsional-diferensial merupakan kontroler yang aksi kontrolnya mempunyai sifat proporsional dan diferensial terhadap sinyal kesalahan. Kontroler PD berfungsi untuk meningkatkan stabilitas sistem , mengurangi overshoot dan meningkatkan transient response.

Untuk kontroler PD, sinyal kesalahan $e(t)$ merupakan masukan kontroler sedangkan keluaran kontroler adalah sinyal kontrol $u(t)$. Hubungan antara masukan kontroler $e(t)$ dan keluaran kontroler $u(t)$ adalah

$$u(t) = K_p \left(e(t) + \tau_d \frac{de(t)}{dt} \right) \quad (2.62)$$

atau dalam besaran transformasi laplace

$$U(s) = K_p (1 + \tau_d s) E(s) \quad (2.63)$$

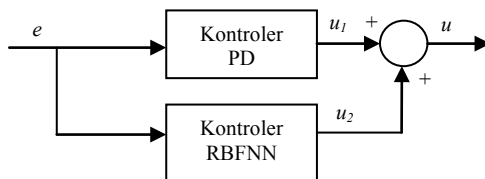
di mana : K_p = penguatan propotional
 τ_d = waktu diferensial

Sehingga fungsi alih kontroler PD adalah

$$\frac{U(s)}{E(s)} = K_p (1 + \tau_d s) \quad (2.64)$$

2.8.4 Kontroler RBFNN – PD Hybrid

Kontroler RBFNN – PD Hybrid merupakan sebuah kontroler yang menggabungkan 2 buah jenis kontroler yaitu kontroler PD dan RBFNN yang dipasang secara paralel sebagaimana ditunjukkan pada Gambar 2.17.

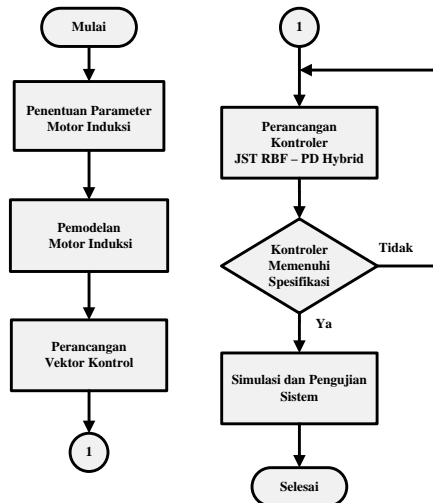


Gambar 2.17 Kontroler RBFNN-PD Hybrid

BAB III

PERANCANGAN SISTEM

Pada bab ini dibahas mengenai perancangan kontroler JST RBF – PD Hybrid untuk pengaturan kecepatan motor induksi tiga fasa di mana vektor kontrol digunakan sebagai metode pengontrolanya. Alur kerja yang dilakukan dalam perancangan ini dapat dijelaskan seperti pada Gambar 3.1.



Gambar 3.1 Diagram alir perancangan sistem

Dari Gambar 3.1 perancangan sistem dibagi menjadi beberapa tahapan yang dilakukan secara runtun. Tahap pertama adalah menentukan parameter parameter motor induksi. Tahap kedua adalah melakukan pemodelan motor induksi berdasarkan model matematis pada bab 2 bagian 2.3. Tahap ketiga adalah merancang metode vektor kontrol berdasarkan diagram blok *indirect vector control* pada Gambar 2.9. Tahap keempat ialah merancang kontroler kecepatan dengan menggunakan JST RBF – PD Hybrid . Tahap terakhir adalah melakukan pengujian kontroler pada motor induksi dengan kondisi tanpa beban dan berbeban. Seluruh perancangan dilakukan menggunakan perangkat lunak program simulasi.

3.1 Penentuan Parameter Motor Induksi Tiga Fasa

Motor induksi yang digunakan dalam perancangan ini adalah motor induksi jenis rotor sangkar (*squirrel-cage*). Pada perancangan simulasi ini digunakan motor induksi tenaga 3 HP/2,4 KW dengan parameter – parameter seperti pada Tabel 3.1 [8].

Tabel 3.1 Parameter – parameter motor induksi

| No. | Parameter | Nilai | Satuan |
|-----|--|-------|-------------------|
| 1. | Tenaga motor | 3/2,4 | HP/KW |
| 2. | Tegangan motor (<i>line to line</i>) | 460 | Volt |
| 3. | Frekuensi | 60 | Hz |
| 4. | Jumlah kutub | 4 | |
| 5. | Tahanan stator | 0,738 | Ohm |
| 6. | Tahanan rotor | 0,451 | Ohm |
| 7. | Induktansi diri stator | 0,014 | Henry |
| 8. | Induktansi diri rotor | 0,012 | Henry |
| 9. | Induktansi mutual | 0,369 | Henry |
| 10. | Momen inersia | 0,025 | kg.m ² |
| 11 | Koefisien redaman | 0,000 | Nm s/rad |

Berdasarkan parameter motor induksi pada Tabel 3.1, maka dengan menggunakan persamaan 2.1, kecepatan medan putar stator (N_s) sebagai berikut:

$$\begin{aligned}
 N_s &= \frac{120 \cdot f}{p} = \frac{120 \cdot 60}{4} \\
 &= 1800 \quad rpm \\
 &= 188,5 \quad rps
 \end{aligned}
 \tag{3.1}$$

Pada saat kondisi tanpa beban nilai kecepatan putar rotor mendekati kecepatan medan putar stator/ kecepatan sinkron ($N_r \approx N_s$) atau dengan kata lain slip bernilai nol. Nilai induktansi stator dan rotor

diperoleh dengan melakukan perhitungan seperti pada persamaan 3.2 dan 3.3.

$$\begin{aligned}
 L_s &= L_{Is} + L_m \\
 &= 0,014 + 0,369 \\
 &= 0,383
 \end{aligned} \tag{3.2}$$

$$\begin{aligned}
 L_r &= L_{Ir} + L_m \\
 &= 0,012 + 0,369 \\
 &= 0,381
 \end{aligned} \tag{3.3}$$

3.2 Pemodelan Motor Induksi Tiga Fasa

Pada tahapan ini motor induksi 3 fasa dibuat dalam d - q model seperti yang telah dijelaskan pada bab 2 bagian 2.3. Berdasarkan persamaan 2.18 maka dapat dinyatakan dalam bentuk persamaan state dengan penurunan sebagai berikut :

$$\begin{aligned}
 \begin{bmatrix} v_{qs} \\ v_{ds} \\ v_{qr} \\ v_{dr} \end{bmatrix} &= \begin{bmatrix} R_s & \omega & 0 & \omega L_m \\ -\omega L_s & R_s & -\omega L_m & 0 \\ 0 & (\omega - \omega_r)L_m & R_r & (\omega - \omega_r)L_r \\ -(\omega - \omega_r)L_m & 0 & -(\omega - \omega_r)L_r & R_r \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix} \\
 &+ \begin{bmatrix} L_s & \omega & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix}
 \end{aligned} \tag{3.4}$$

di mana : v_{qr} dan $v_{dr} = 0$

atau dapat ditulis :

$$v_{dq} = A i_{dq} + B \dot{i}_{dq} \tag{3.5}$$

di mana :

$$A = \begin{bmatrix} R_s & \omega & 0 & \omega L_m \\ -\omega L_s & R_s & -\omega L_m & 0 \\ 0 & (\omega - \omega_r)L_m & R_r & (\omega - \omega_r)L_r \\ -(\omega - \omega_r)L_m & 0 & -(\omega - \omega_r)L_r & R_r \end{bmatrix}$$

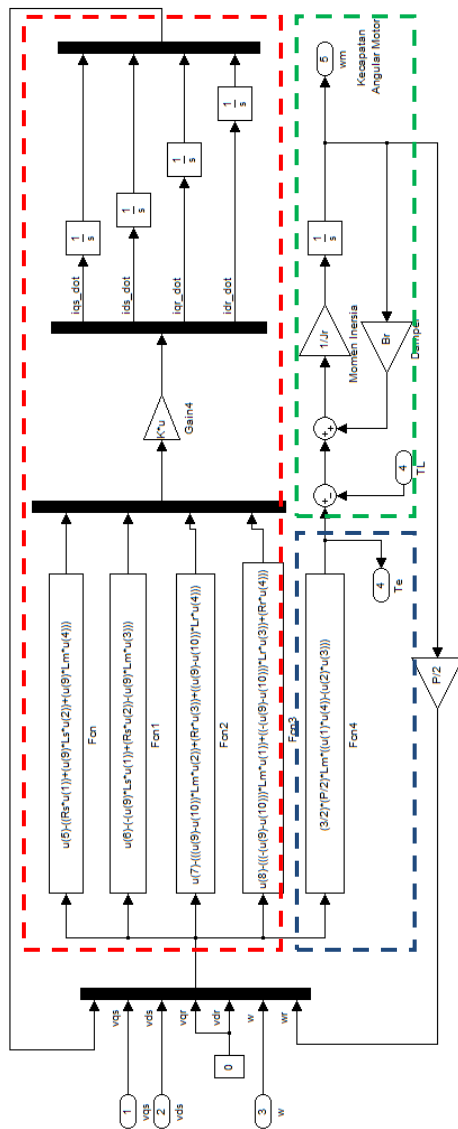
$$B = \begin{bmatrix} L_s & \omega & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix}$$

sehingga didapatkan persamaan state :

$$\begin{aligned} \dot{i}_{dq} &= -B^{-1}Ai_{dq} + B^{-1}V_{dq} \\ &= B^{-1}(Ai_{dq} + V_{dq}) \end{aligned} \quad (3.6)$$

Berdasarkan persamaan 3.6 , 2.20 dan persamaan 2.22 maka blok perancangan simulasi yang dibuat ditunjukkan pada Gambar 3.2 dengan keterangan sebagai berikut :

- - - - = Pemodelan motor induksi 3 fasa (persamaan 3.6)
- - - - = Perhitungan torsi yang dihasilkan (persamaan 2.14)
- - - - = Perhitungan kecepatan (persamaan 2.15)



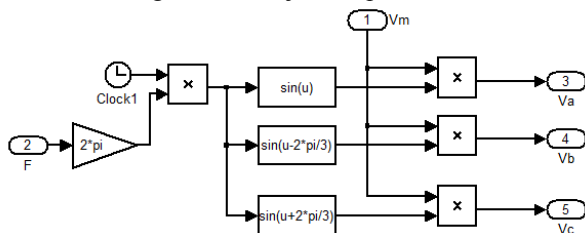
Gambar 3.2 Blok simulasi pemodelan motor induksi tiga fasa

Hasil pemodelan motor induksi kemudian diuji menggunakan parameter-parameter yang terdapat pada Tabel 3.1 dan sumber tegangan tiga fasa secara langsung. Sumber tiga fasa dibuat berdasarkan persamaan 3.7.

$$\begin{aligned}v_{as} &= v_m \sin(\omega t) \\v_{bs} &= v_m \sin(\omega t - 2\pi/3) \\v_{cs} &= v_m \sin(\omega t + 2\pi/3)\end{aligned}\quad (3.7)$$

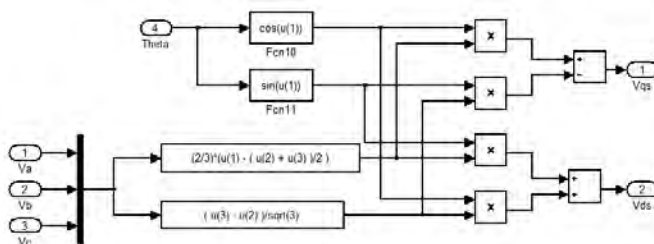
di mana : $\omega = 2\pi f$

Blok simulasi tiga fasa ditunjukkan pada Gambar 3.3.



Gambar 3.3 Blok simulasi sumber 3 fasa

Sumber harus ditransformasi kedalam koordinat rotasi 2 fasa menggunakan persamaan 2.6 dan 2.9 untuk menghubungkan sumber dengan motor induksi. Blok simulasi transformasi yang dibuat, ditunjukkan pada Gambar 3.4.



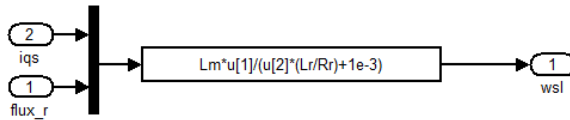
Gambar 3.4 Blok simulasi transformasi a-b-c ke d-q

3.3 Perancangan Vektor Kontrol

Pada tahapan ini metode vektor kontrol dirancang untuk pengontrolan motor induksi. Jenis vektor kontrol yang digunakan ialah jenis *indirect vector control*. Berdasarkan blok diagram pada Gambar 2.9 maka dapat dibagi menjadi beberapa bagian atau sub sistem yaitu perhitungan kecepatan slip (ω_{sl}), perhitungan arus i_{qs}^* , perhitungan arus i_{ds}^* , unit vektor generator, transformasi vektor, *hysterisis band current control*, dan inverter.

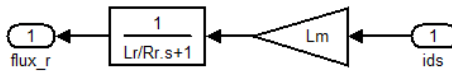
3.3.1 Perancangan Perhitungan Kecepatan Slip (ω_{sl})

Perhitungan kecepatan slip ini didapatkan berdasarkan persamaan 2.31. Gambar 3.5 menunjukkan blok simulasi perhitungan kecepatan slip yang dibuat.



Gambar 3.5 Blok simulasi perhitungan kecepatan slip

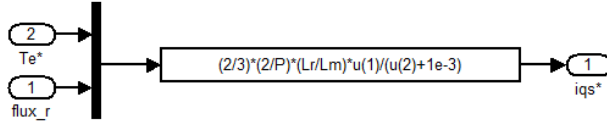
di mana nilai fluks λ_r didapatkan dari persamaan 2.29, sehingga blok simulasi perhitungan fluks (λ_r) yang dibuat dapat ditunjukkan seperti pada Gambar 3.6.



Gambar 3.6 Blok simulasi perhitungan fluks

3.3.2 Perancangan Perhitungan Arus i_{qs}^*

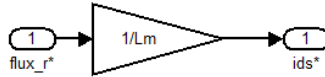
Arus i_{qs}^* merupakan arus referensi yang digunakan untuk mengatur kecepatan. Nilai arus ini didapatkan dari keluaran kontroler kecepatan yaitu sinyal kontrol berupa nilai torsi referensi (T_e) yang kemudian diubah menjadi nilai arus referensi (i_{qs}^*) dengan menggunakan persamaan 2.32. Gambar 3.7 menunjukkan blok simulasi perhitungan arus i_{qs}^* .



Gambar 3.7 Blok simulasi perhitungan arus i_{qs}^*

3.3.3 Perancangan Perhitungan Arus i_{ds}^*

Arus i_{ds}^* merupakan arus referensi yang digunakan untuk mengatur fluks. Nilai arus ini didapatkan dari nilai fluks referensi yang diberikan (λ_r^* konstan) yang kemudian diubah menjadi nilai arus referensi (i_{ds}^*) dengan menggunakan persamaan 2.30. Gambar 3.8 menunjukkan blok simulasi perhitungan arus i_{ds}^* .



Gambar 3.8 Blok simulasi perhitungan arus i_{ds}^*

3.3.4 Perancangan Unit Vektor Generator

Unit vektor generator merupakan bagian dari sistem yang berfungsi untuk menghasilkan komponen dari unit vektor ($\sin \theta$ dan $\cos \theta$) yang digunakan untuk proses transformasi vektor. Unit vektor didapatkan dengan cara mendapatkan nilai θ dengan menggunakan persamaan 3.8.

$$\theta = \theta_r + \theta_{sl} \quad (3.8)$$

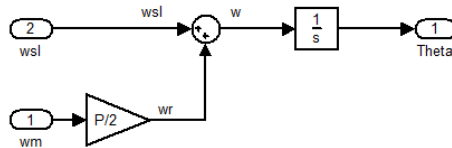
Sehingga dengan menurunkan persamaan 3.8 di dapatkan persamaan sebagai berikut :

$$\omega = \omega_r + \omega_{sl} \quad (3.9)$$

dengan

$$\omega_r = \frac{p}{2} \omega_m \quad (3.10)$$

Berdasarkan persamaan 3.9 maka blok simulasi yang dibuat, ditunjukkan pada Gambar 3.9.

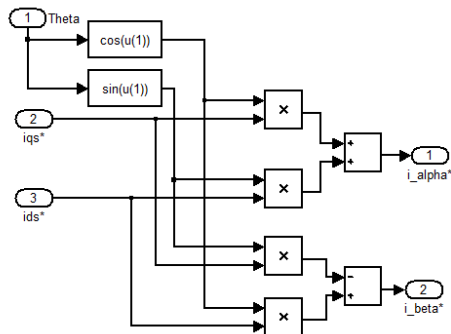


Gambar 3.9 Blok simulasi unit vektor generator

3.3.5 Perancangan Transformasi Vektor

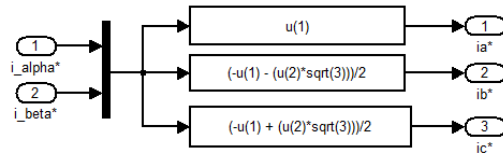
Transformasi vektor digunakan untuk merubah sistem koordinat 3 fasa ke koordinat rotasi 2 fasa atau sebaliknya. Berdasarkan blok diagram pada Gambar 2.9 transformasi vektor digunakan untuk merubah arus referensi 2 fasa (i_{ds}^* dan i_{qs}^*) menjadi 3 fasa (i_a^* , i_b^* dan i_c^*). Untuk merubah nya dilakukan dengan 2 tahap yaitu :

- Merubah dari sistem koordinat rotasi 2 fasa menjadi sistem koordinat stationer 2 fasa atau disebut transformasi Park dengan menggunakan persamaan 2.8. Gambar 3.10 menunjukkan blok simulasi transformasi Park yang dibuat.



Gambar 3.10 Blok simulasi transformasi Park ($d-q$ ke $\alpha-\beta$)

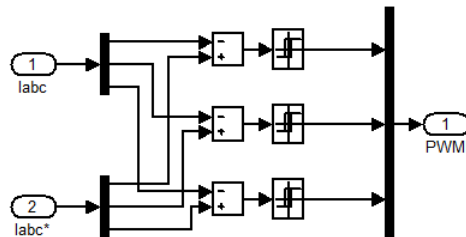
- Merubah dari sistem koordinat stationer 2 fasa menjadi sistem koordinat stationer 3 fasa atau disebut transformasi Clarke dengan menggunakan persamaan 2.7. Gambar 3.11 menunjukkan blok simulasi transformasi Clarke yang dibuat.



Gambar 3.11 Blok simulasi transformasi Clarke (α - β ke a - b - c)

3.3.6 Perancangan *Hysterisis Band Current Control*

Hysterisis band current control berfungsi untuk menghasilkan sinyal PWM sebagai sinyal kontrol untuk mengatur nilai masukan motor induksi tiga fasa melalui inverter. Gambar 3.12 menunjukkan blok simulasi yang dibuat.



Gambar 3.12 Blok simulasi *hysterisis band current control*

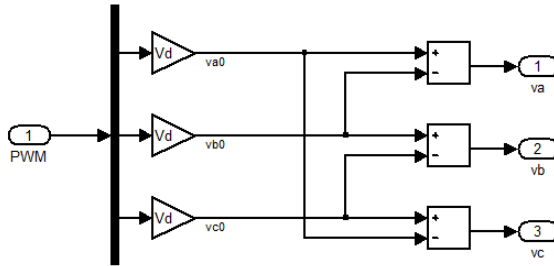
di mana : i_{abc} = arus aktual motor
 i_{abc}^* = arus referensi motor

3.3.7 Perancangan Inverter

Pada perancangan ini inverter berfungsi merubah sinyal PWM dari *hysterisis band current control* menjadi sumber 3 fasa sebagai masukan motor induksi. Blok simulasi inverter pada perancangan ini dibuat sederhana dengan tidak merubah fungsi inverter pada sistem yang dibuat. Pemodelan inverter ini didasarkan pada persamaan 3.11,

$$\begin{aligned} V_a &= V_{a0} - V_{b0} \\ V_b &= V_{b0} - V_{c0} \\ V_c &= V_{c0} - V_{a0} \end{aligned} \quad (3.11)$$

sehingga blok simulasi inverter yang dibuat ditunjukkan pada Gambar 3.13.

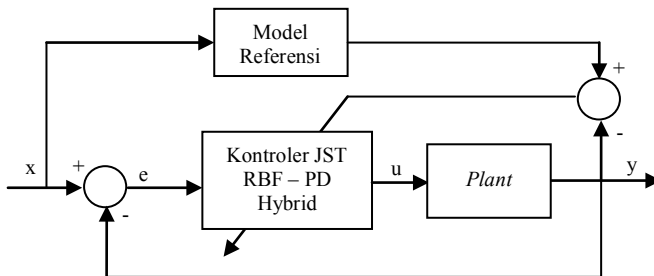


Gambar 3.13 Blok simulasi inverter

di mana : V_d = tegangan *line to line*

3.4 Perancangan Kontroler

Pada tugas ini JST RBF – PD Hybrid digunakan sebagai kontroler. Gambar 3.14 menunjukkan diagram blok penggunaan kontroler pada sistem yang dibuat.



Gambar 3.14 Diagram blok sistem dengan kontroler JST RBF – PD Hybrid

Sebagaimana telah dijelaskan pada bab 2 bagian 2.8.4 kontroler JST RBF – PD Hybrid merupakan kontroler yang menggabungkan 2 buah jenis kontroler yaitu kontroler PD dan JST RBF seperti ditunjukkan pada Gambar 2.17. JST-RBF disini berfungsi sebagai integrator dengan

tujuan untuk menghilangkan *steady state error*. Model referensi merupakan model respon yang diinginkan dari keluaran *plant*.

3.4.1 Perancangan Kontroler JST RBF

JST RBF pada kontroler disini digunakan sebagai integrator di mana struktur RBF dibentuk berdasarkan kontroler integral diskrit dengan penurunan sebagai berikut:

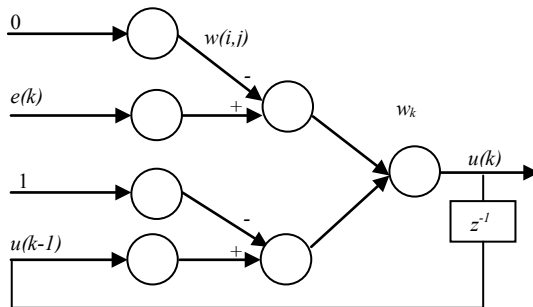
$$u(k) = Ki \sum_{i=1}^k e(i) \quad (3.12)$$

$$u(k-1) = Ki \sum_{i=1}^{k-1} e(i) \quad (3.13)$$

Jika persamaan 3.12 dikurang dengan persamaan 3.13 maka akan dihasilkan

$$u(k) = K_i e(k) + u(k-1) \quad (3.14)$$

Berdasarkan persamaan 3.14 maka dengan menggunakan konfigurasi seperti yang telah dijelaskan pada bab 2 bagian 2.7, maka dapat dibentuk JST RBF dengan fungsi aktivasi gaussian seperti pada Gambar 3.15.



Gambar 3.15 Struktur JST RBF

Struktur yang dibuat terdiri dari 4 masukan yaitu $e(k)$, $u(k-1)$, faktor pengali nilai tengah $e(k)$ dan faktor pengali nilai tengah $u(k-1)$. Karena pada sistem yang dibuat diinginkan *error* disekitar 0 maka agar nilai tengah bernilai 0, faktor pengali nilai tengah $e(k)$ dibuat 0. Pada neuron *output* terdapat 2 masukan dengan bobot masing-masing mewakili koefisien pada persamaan 3.14 sehingga bobot pada masukan 1 neuron *output* ialah nilai K_i dan bobot masukan 2 neuron *output* ialah 1.

3.4.2 Algoritma Pelatihan JST RBF

Proses pembelajaran dilakukan secara online atau pembelajaran langsung pada *plant*. Pada algoritma ini metode *backpropagation* diterapkan dalam proses pelatihanya. Berikut algoritma pelatihan yang dilakukan :

- a. Langkah 1 : Menentukan model hasil desain yang diinginkan dalam bentuk orde 1
- b. Langkah 2 : Menentukan α (*learning rate*) yang besarnya antara $0 < \alpha < 1$ serta menentukan λ (*gain rate*).
- c. Langkah 3 : Menentukan banyak *input*, *hidden* dan *output* (*input* = 4, *hidden* = 2, *output* = 1)
- d. Langkah 4 : Memberikan nilai bobot awal secara random.
- e. Langkah 5 : Melakukan perhitungan maju / *feed forward* pada lapisan *hidden*. Berikut ini merupakan perhitungan yang dilakukan.

1) *Input* 1 dan 2 ke neuron *hidden* 1

$$z_i^{h_1} = w_{21}x_1 - w_{11}x_2 \quad (3.15)$$

$$y_i^{h_1} = e^{-(z_i^{h_1})^2} \quad (3.16)$$

2) *Input* 3 dan 4 ke neuron *hidden* 2

$$z_i^{h_2} = w_{32}x_3 - w_{22}x_4 \quad (3.17)$$

$$y_i^{h_2} = e^{-(z_i^{h_2})^2} \quad (3.18)$$

- f. Langkah 6 : Melakukan perhitungan maju pada lapisan *output*. Berikut ini merupakan perhitungan yang dilakukan.

$$z_p^o = \sum_{k=1}^2 w_k y_i^{h_k} \quad (3.19)$$

$$y_p^o = \lambda \cdot z_p^o \quad (3.20)$$

- g. Langkah 7 : Melakukan *backward* yaitu menghitung perbaikan pembobot.

- 1) Menghitung *error* model

$$e_m = y_m - y_p \quad (3.21)$$

di mana : $e_m = \text{error model}$

$y_m = \text{output model}$

$y_p = \text{output plant}$

- 2) Menghitung perambatan *error*

Pada tugas akhir ini nilai *error* dibuat proposional dengan *plant* sehingga untuk mengkoreksi bobot dari *hidden* ke *output* ialah :

$$er = e_m \cdot \frac{1}{k} \quad (3.22)$$

di mana : $k = \text{gain dari plant}$

dan untuk mengkoreksi bobot dari *input* ke *hidden* ialah :

$$err_i = \frac{w_k y_i^{h_k}}{z_p^o} \quad (3.23)$$

- 3) Revisi bobot dari *hidden* ke *output*

Perhitungan revisi bobot dari *hidden* ke *output* dilakukan dengan menggunakan persamaan 3.18

$$w_k(\text{baru}) = w_k(\text{lama}) + er \cdot f'(z_p^o) \cdot y_i^{h_k} \quad (3.24)$$

di mana $f'(z_0^p) =$ turunan fungsi aktivasi *linear*

- 4) Revisi bobot dari *input* ke *hidden*
Perhitungan revisi bobot dari *input* ke *hidden* dilakukan dengan menggunakan persamaan 3.19

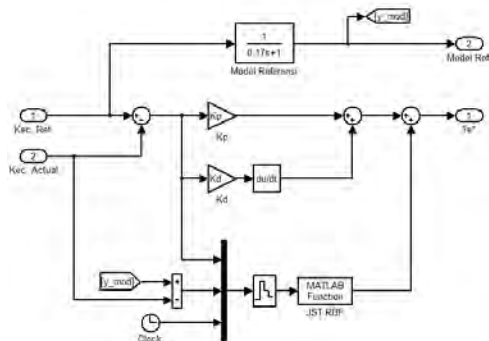
$$w_{ij}(\text{baru}) = w_{ij}(\text{lama}) + \text{err}_i \cdot f'(z_i^{h_k}) \cdot y_i^{h_k} \quad (3.25)$$

di mana $f'(z_i^{h_k}) =$ turunan fungsi aktivasi gaussian

- 5) Revisi elemen *input*
Pada tahapan ini dilakukan perbaharuan terhadap nilai *input* yaitu $e(k)$ dan $u(k-1)$.
- h. Langkah 8 : dilakukan pengujian terhadap *output* yang dihasilkan dan proses dilakukan secara berulang.

3.4.3 Perancangan Kontroler JST RBF – PD Hybrid

Berdasarkan diagram blok pada Gambar 2.21 maka kontroler yang dirancang terdiri dari 2 jenis yaitu kontroler PD dan JST RBF, sehingga blok simulasi yang dibuat dapat ditunjukkan pada Gambar 3.16



Gambar 3.16 Blok simulasi kontroler JST RBF – PD Hybrid

Pada perancangan ini nilai dari parameter K_p dan K_d di dapatkan dengan cara penelahan atau *tunning* secara manual.

3.5 Perhitungan Beban Nominal Motor Induksi

Beban nominal adalah beban yang diterima atau ditanggung oleh motor sehingga motor masih dapat bekerja pada kecepatan nominalnya. Pada tugas akhir ini beban nominal di dapatkan dari perhitungan daya *output* atau secara matematis dapat ditulis seperti pada persamaan berikut:

$$P_{out} = P_{tot} - P_{op} \quad (3.26)$$

$$P_{out} = T \cdot \omega_m \quad (3.27)$$

di mana : P_{tot} = Daya total *output* motor

P_{out} = Daya *output* motor

P_{op} = Daya *output* operasional

T = Torsi beban nominal

Kecepatan nominal motor (ω_m) merupakan kecepatan referensi yang digunakan untuk pengujian motor. Pada pengujian yang dilakukan digunakan nilai referensi sebesar 150 rps atau 1432,4 rpm.

Daya yang di keluarkan oleh motor selama motor beroperasi (P_{op}) di dapatkan dengan menggunakan persamaan 3.21.

$$P_{op} = \frac{1}{2} J \omega_m \quad (3.28)$$

Berdasarkan parameter pada Tabel 3.1 maka dapat dihitung beban nominal dengan menggunakan persamaan 3.26, 3.27 dan 3.28 sehingga didapatkan nilai torsi beban nominal dengan nilai kecepatan referensi sebesar 150 rps ialah sebesar 14,13 Nm.

BAB IV

HASIL SIMULASI DAN ANALISIS SISTEM

Pada bab ini dibahas mengenai simulasi dan analisa data hasil pengujian terhadap sistem yang telah dibuat. Pengujian dibagi menjadi pengujian *open loop*, pengujian simulasi vektor kontrol dengan kontroler PI dan pengujian simulasi vektor kontrol dengan kontroler JST RBF - PD Hybrid.

Pada pengujian simulasi vektor kontrol dengan kontroler PI dan pengujian vektor kontrol dengan kontroler JST RBF - PD Hybrid, pengujian dilakukan dengan cara memberikan nilai *setpoint* kecepatan sebesar 150 rps dan kemudian dilakukan pembebanan pada motor induksi. Beban yang diberikan adalah beban langsung. Pemberian beban langsung pada pengujian dilakukan dengan 2 cara yaitu:

- a. Pembebanan dilakukan secara langsung sesuai dengan nilai beban yang ditentukan.
- b. Pembebanan dilakukan secara bertahap/*gradual* hingga mencapai nilai beban yang ditentukan dengan rentang waktu tertentu.

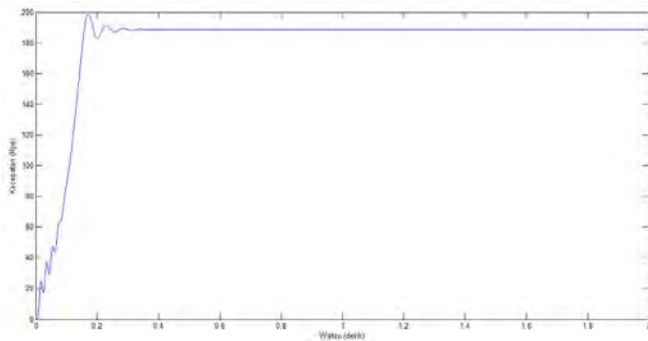
Nilai beban yang diberikan terdiri dari 3 jenis yaitu beban minimal, beban nominal dan beban maksimal. Beban minimal didapatkan dari 20% dikurangi beban nominalnya sehingga didapatkan nilai sebesar 11,44 Nm. Beban maksimal didapatkan dari 20% ditambah beban nominalnya sehingga didapatkan nilai sebesar 17,16 Nm.

4.1 Simulasi *Open Loop* Motor Induksi

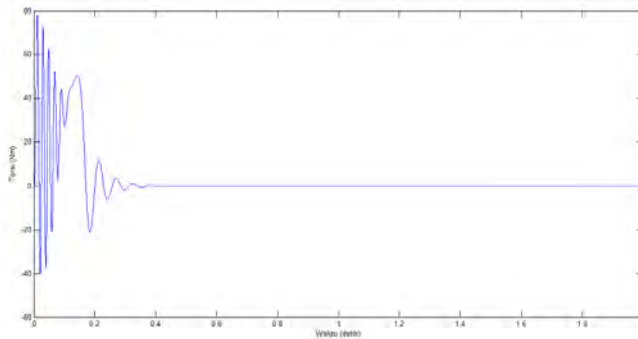
Pada bagian ini dilakukan pengujian motor induksi secara *open loop* dengan masukan yang diberikan berupa sumber tiga fasa secara langsung dan tanpa beban (beban = 0). Hasil pengujian yang dilakukan ditunjukkan pada Gambar 4.1 dan 4.2.

Berdasarkan hasil respon kecepatan yang dihasilkan, motor induksi mencapai keadaan tunak pada $t = 0,35$ detik dengan nilai kecepatan sama dengan kecepatan sinkronnya yaitu sebesar 188,5 rps. Pada gambar 4.1 dapat dilihat terjadi *overshoot* dengan nilai *maximum overshoot* (M_p) sebesar 5,35 % dari keadaan tunaknya.

Pada gambar 4.2 menunjukkan nilai torsi *starting* sebesar 77,35 Nm pada saat $t = 0,01$ detik. Pada saat respon kecepatan berada pada keadaan tunak nilai torsi berada sekitar nilai nol. Hal ini dikarenakan motor dioperasikan pada kondisi tanpa beban.



Gambar 4.1 Respon kecepatan motor induksi



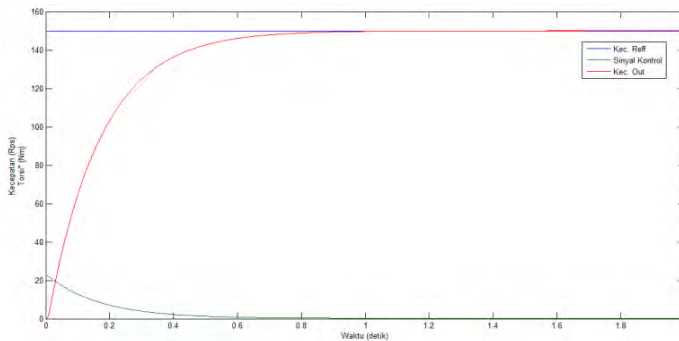
Gambar 4.2 Respon torsi motor induksi

4.2 Simulasi Vektor Kontrol dengan Kontroler PI

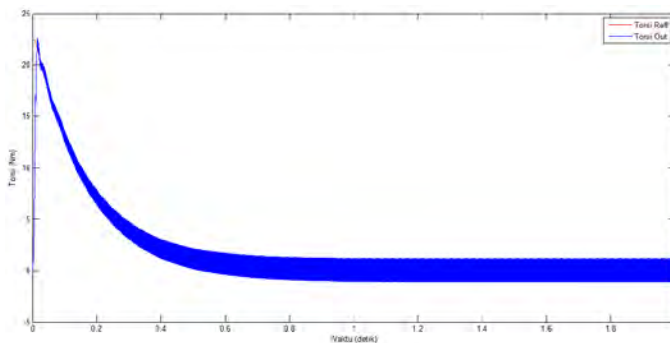
Pada bagian ini dilakukan pengujian terhadap simulasi vektor kontrol dengan kontroler kecepatan yang digunakan ialah kontroler PI. Sinyal kontrol yang dihasilkan berupa nilai torsi referensi dan untuk nilai fluks referensi diberi nilai konstan sebesar 0,7 weber. Nilai parameter kontroler yang diberikan ialah $K_p=0,15$ dan $K_i = 0,0085$.

4.2.1 Pengujian Tanpa Beban

Respon yang dihasilkan dari pengujian tanpa beban (beban=0) pada simulasi vektor kontrol dengan kontroler PI ditunjukkan pada Gambar 4.3 dan Gambar 4.4.



Gambar 4.3 Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler PI tanpa beban



Gambar 4.4 Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler PI tanpa beban

Berdasarkan hasil respon kecepatan yang di dapatkan maka dapat dilakukan analisa karakteristik respon dengan menggunakan pendekatan respon orde 1, sehingga didapatkan karakteristik kerja dari sistem seperti tampak pada Tabel 4.1.

Tabel 4.1 Karakteristik respon kecepatan motor induksi menggunakan vektor kontrol dengan kontroler PI tanpa beban

| Karakteristik | Nilai | Keterangan |
|-------------------------|-------|---|
| Konstanta waktu (detik) | 0,17 | $\omega = (0,632).150 = 94,8 \text{ rps}$ |
| Settling time (detik) | 0,85 | $t_s(\pm 0,5\%)$ |

Tabel 4.1 Karakteristik respon kecepatan motor induksi menggunakan vektor kontrol dengan kontroler PI tanpa beban (Lanjutan)

| Karakteristik | Nilai | Keterangan |
|-------------------------------|-------|-------------------|
| <i>Rise time</i> (detik) | 0,50 | $t_s(5\% - 95\%)$ |
| <i>Delay time</i> (detik) | 0,12 | t_d |
| <i>Error Steady state</i> (%) | 0,00 | e_{ss} |

Pada Gambar 4.3 dapat dilihat hubungan sinyal kontrol dengan respon kecepatan bahwa sinyal kontrol yang diberikan berbanding terbalik dengan respon kecepatan yang dihasilkan. Nilai yang diberikan pada saat *starting* berada pada nilai 22,5 Nm dan mengalami penurunan hingga mencapai nilai di sekitar 0 Nm pada waktu $t = 0,8$ detik.

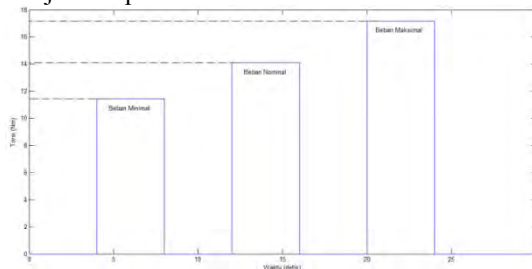
Respon torsi menunjukkan pada saat awal, torsi mengejar nilai referensi yang diberikan dan pada waktu sekitar 0,012 detik torsi yang dihasilkan mengikuti nilai referensi torsi yang diberikan dengan nilai 21,7 Nm. Berdasarkan Gambar 4.4 dapat dilihat pula terdapat *ripple* pada torsi yang dihasilkan sebesar 2,3 Nm.

4.2.2 Pengujian dengan Beban

Sebagaimana yang telah dijelaskan sebelumnya pengujian dengan beban dilakukan dengan 2 cara yaitu beban secara langsung dan *gradual*. Nilai beban yang diberikan terdiri dari beban minimal, nominal dan *gradual*.

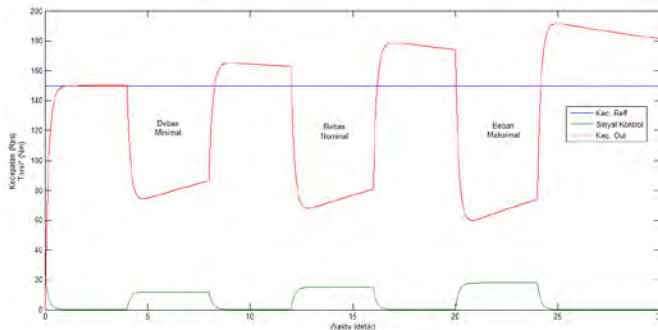
4.2.2.1 Beban secara langsung

Pada pengujian ini beban diberikan secara langsung dengan lama pemberian beban untuk setiap jenisnya (minimal, nominal dan maksimal) ialah selama 4 detik. Representasi nilai beban yang diberikan ditunjukkan pada Gambar 4.5.



Gambar 4.5 Representasi pemberian beban secara langsung

Respon yang dihasilkan dari pengujian menggunakan beban secara langsung pada simulasi vektor kontrol dengan kontroler PI ditunjukkan pada Gambar 4.6 dan Gambar 4.7.



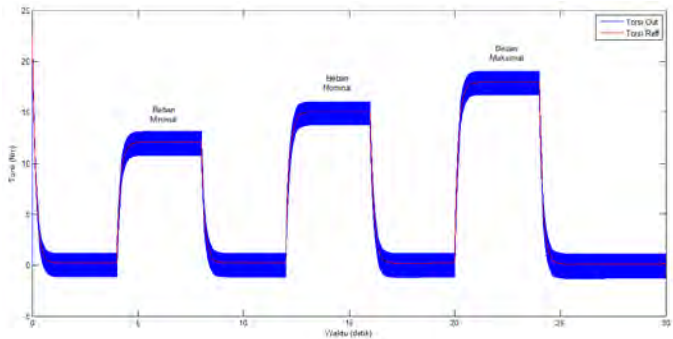
Gambar 4.6 Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler PI dan beban langsung

Berdasarkan hasil pengujian, ketika diberi beban terjadi penurunan kecepatan untuk setiap jenis beban yang diberikan. Setelah kecepatan turun, pada waktu $t = 4,71$ detik (minimal), $t = 12,75$ detik (nominal), $t = 20,86$ detik (maksimal) kecepatan mulai naik perlahan, namun dalam waktu 4 detik kecepatan tidak dapat mengejar nilai referensinya.

Kemudian setelah beban dilepas kondisi yang terjadi ialah berbanding terbalik dengan kondisi saat diberi beban yaitu kecepatan mengalami kenaikan hingga melebihi nilai referensinya. Setelah kecepatan naik, pada waktu $t = 8,76$ detik (minimal), $t = 16,78$ detik (nominal), $t = 24,81$ detik (maksimal) kecepatan mulai turun perlahan, namun dalam waktu 4 detik kecepatan tidak dapat mengejar nilai referensinya.

Pada Gambar 4.6 dapat dilihat hubungan antara sinyal kontrol dengan respon kecepatan bahwa ketika diberi beban kecepatan tidak masuk pada kondisi tunaknya meskipun nilai sinyal kontrol yang diberikan telah sama dengan torsi beban yang diberikan. Lama waktu saat mulai diberi beban hingga sinyal kontrol mencapai nilai torsi beban ialah sama dengan lama waktu penurunan kecepatan yang terjadi yaitu

rata-rata sekitar 0,77 detik dan begitu pula saat beban di lepas yaitu rata-rata sekitar 0,78 detik.

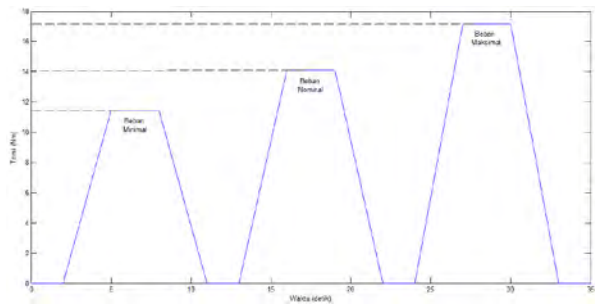


Gambar 4.7 Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler PI dan beban langsung

Respon torsi pada Gambar 4.7 menunjukkan pada saat terjadi perubahan beban, torsi mengikuti nilai referensi yang diberikan, namun terdapat *ripple* sebesar 2,3 Nm.

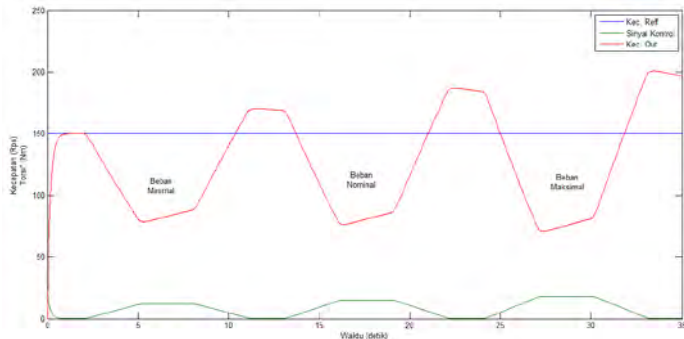
4.2.2.2 Beban secara gradual

Pada pengujian ini beban diberikan secara *gradual* dengan nilai beban diberikan secara bertahap dari 0 sampai dengan nilai beban yang telah ditentukan selama 3 detik. Setelah itu motor diberi beban untuk setiap jenisnya (minimal, nominal dan maksimal) selama 3 detik. Representasi nilai beban yang diberikan ditunjukkan pada gambar 4.8.

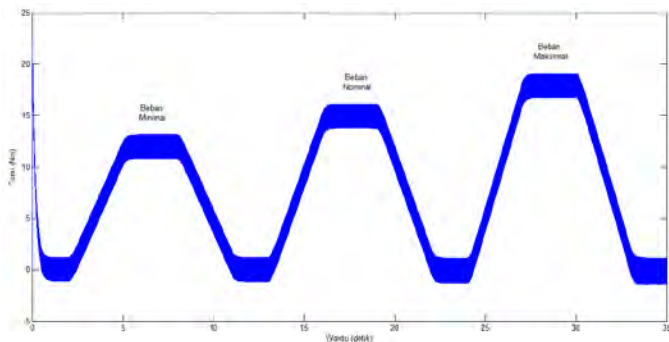


Gambar 4.8 Representasi pemberian beban secara *gradual*

Respon yang dihasilkan dari pengujian menggunakan beban secara *gradual* pada simulasi vektor kontrol dengan kontroler PI ditunjukkan pada Gambar 4.9 dan Gambar 4.10.



Gambar 4.9 Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler PI dan beban *gradual*



Gambar 4.10 Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler PI dan beban *gradual*

Berdasarkan hasil pengujian, ketika beban diberikan secara bertahap dari 0 sampai dengan nilai beban yang ditentukan terjadi penurunan kecepatan untuk setiap jenis beban yang diberikan dengan selang waktu selama 3 detik. Setelah kecepatan turun, kecepatan mulai naik perlahan, namun dalam waktu 3 detik kecepatan tidak dapat mengejar nilai referensinya.

Kemudian ketika beban dilepas secara perlahan (berbeban – beban = 0) kecepatan mengalami kenaikan hingga melebihi nilai referensinya dengan selang waktu 3 detik. Setelah kecepatan naik kecepatan mulai turun perlahan, namun dalam waktu 3 detik kecepatan tidak dapat mengejar nilai referensinya.

Pada Gambar 4.9 dapat dilihat hubungan antara sinyal kontrol dengan respon kecepatan memiliki karakteristik yang sama dengan saat diberi beban secara langsung yaitu nilai sinyal kontrol yang dihasilkan mengikuti nilai beban yang diberikan namun kecepatan tetap tidak mencapai keadaan tunaknya. Lama waktu saat mulai diberi beban hingga sinyal kontrol mencapai nilai torsi beban ialah sama dengan lama waktu penurunan kecepatan yang terjadi dan begitu pula saat beban diturunkan yaitu selama 3 detik.

Respon torsi pada gambar 4.10 menunjukkan pada saat terjadi perubahan beban nilai torsi yang dihasilkan mengikuti torsi mengikuti nilai referensi yang diberikan (sinyal kontrol pada Gambar 4.9), namun terdapat *ripple* sebesar 2,3 Nm.

4.3 Simulasi Vektor Kontrol dengan Kontroler JST RBF – PD Hybrid

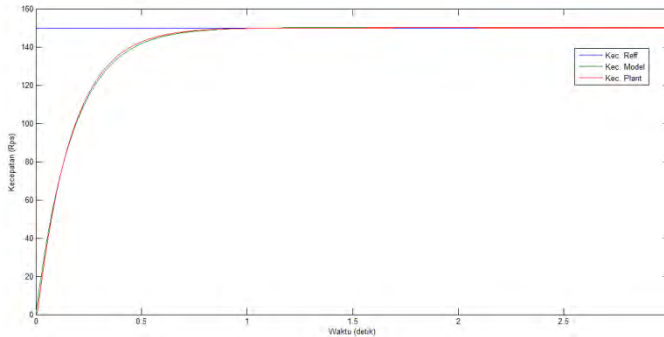
Pada bagian ini dilakukan pengujian terhadap simulasi vektor kontrol dengan menggunakan JST RBF – PD Hybrid sebagai kontroler kecepatan. Untuk nilai fluks referensi diberi nilai konstan sebesar 0,7 weber. Pada simulasi ini terdapat beberapa prosedur pengujian yang dilakukan yaitu :

4.3.1 Proses Pembelajaran

Pada kontroler jaringan syaraf tiruan perlu dilakukan proses *learning*/pembelajaran terlebih dahulu untuk mendapatkan nilai bobot awal sehingga sistem yang dibuat perlu dicari pemodelan sistem dalam bentuk fungsi alih. Proses mendapatkan fungsi alih dilakukan dengan menggunakan identifikasi parameter total. Fungsi alih yang didapat ialah sebagai berikut :

$$\frac{Y(s)}{U(s)} = \frac{38,6s + 3578,4}{s^2 + 93,89s + 4,618}$$

Perbandingan respon hasil pemodelan dan sistem sebenarnya dapat dilihat pada gambar 4.5.



Gambar 4.11 Respon kecepatan motor induksi hasil pemodelan dan *real plant*

Untuk mengetahui validitas hasil pemodelan yang dibuat maka digunakan rumus *RRMSE* (*Relative Root Mean Square Error*) dengan waktu pengamatan 3 detik dan waktu sampling 0,01 detik, sehingga didapatkan :

$$RRMSE = \sqrt{\frac{T}{T_s} \sum_{i=1}^n \frac{y_i - y'_i}{y}} \cdot 100\% = 0,05\%$$

di mana n = jumlah data

y_i = nilai data *real plant*

y'_i = nilai data hasil pemodelan

T = waktu pengamatan

T_s = waktu *sampling*

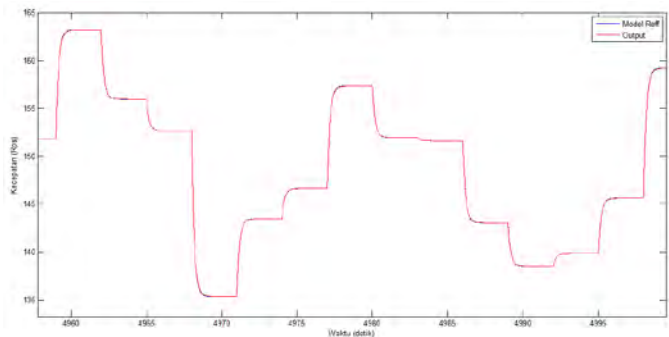
Setelah mendapatkan fungsi alih kemudian ditentukan model referensi. Model referensi adalah model yang diinginkan atau model yang responnya memenuhi spesifikasi karakteristik yang diinginkan dari sistem secara keseluruhan. Model referensi yang digunakan memiliki spesifikasi sebagai berikut :

- a. $t_r = 0,5$ detik
- b. $t_s = 0,85$ detik
- c. $M_p = 0\%$

Maka fungsi alih model referensi yang digunakan ialah :

$$\frac{Y(S)}{X(S)} = \frac{1}{0,17s + 1}$$

Tahapan selanjutnya ialah melakukan learning terhadap pemodelan yang didapatkan dengan masukan berupa nilai *random*. Nilai *random* yang diberikan berkisar antara $\pm 10\%$ dari nilai referensinya. Hasil proses learning ditunjukkan pada Gambar 4.12.



Gambar 4.12 Proses *learning* kontroler

Gambar 4.12 menunjukkan hasil learning dengan waktu pengamatan sekitar 5000 detik dengan nilai parameter yang digunakan ialah $K_p = 0,2$ dan $K_d = 0,0075$. Hasil yang didapat menunjukkan bahwa output pemodelan plant mengikuti model referensi yang diberikan. Dari proses learning didapatkan nilai bobot seperti tampak pada Tabel 4.2.

Tabel 4.2 Nilai bobot hasil *learning*

| Pembobot | Hidden 1 | Hidden 2 | Keterangan |
|----------|----------|----------|--|
| w_1 | 0,054 | 5,093 | Nilai bobot $e(k)$ dan $u(k-1)$ |
| w_2 | 0,000 | 5,409 | Nilai bobot faktor pengali nilai tengah gaussian |
| w_k | 0.192 | 1,000 | Nilai bobot <i>hidden</i> ke <i>output</i> |

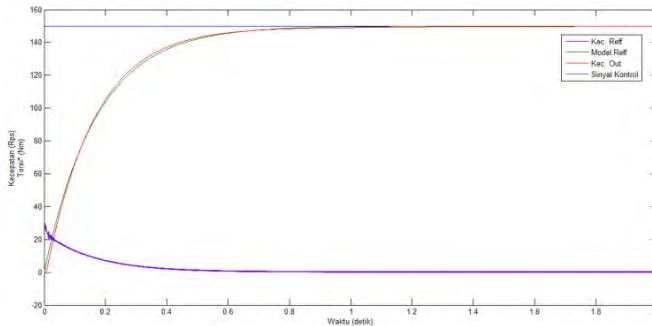
Pada proses *learning* terdapat parameter –parameter JST RBF yang diatur, nilai-nilai parameter hasil proses learning ditunjukkan pada Tabel 4.3.

Tabel 4.3 Parameter JST RBF

| Parameter | Nilai | Keterangan |
|-----------|-------|----------------------|
| α | 1 | <i>Learning rate</i> |
| epp | 10 | Siklus belajar |

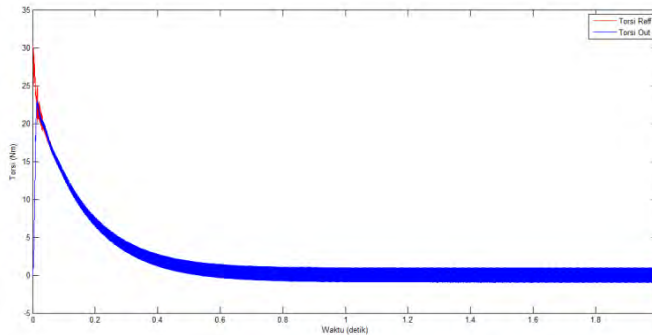
4.3.2 Pengujian Tanpa Beban

Pada bagian ini dilakukan pengujian terhadap simulasi vektor kontrol dan kontroler kecepatan yang digunakan ialah JST RBF – PD Hybrid. Sama halnya dengan kontroler PI, sinyal kontrol yang dihasilkan berupa nilai torsi referensi dan nilai fluks referensi diberi nilai konstan sebesar 0,7 weber. Respon yang dihasilkan dari pengujian tanpa beban (beban = 0) pada simulasi vektor kontrol dengan kontroler PI ditunjukkan pada pada Gambar 4.13 dan Gambar 4.14.



Gambar 4.13 Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid tanpa beban

Berdasarkan respon kecepatan pada Gambar 4.13 maka dapat dilihat bahwa respon yang dihasilkan mengikuti model referensinya. Respon mulai memasuki keadaan tunak (t_s) pada saat $t = 0,85$ detik dengan kecepatan berkisar pada 150 rps.



Gambar 4.14 Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid tanpa beban

Dari respon yang di dapat dilihat pula hubungan sinyal kontrol dengan respon kecepatan bahwa sinyal kontrol yang diberikan berbanding terbalik dengan respon kecepatan yang dihasilkan. Nilai yang diberikan pada saat *starting* berada pada nilai 30 Nm dan mengalami penurunan hingga mencapai nilai di sekitar 0 Nm pada waktu $t = 0,8$ detik.

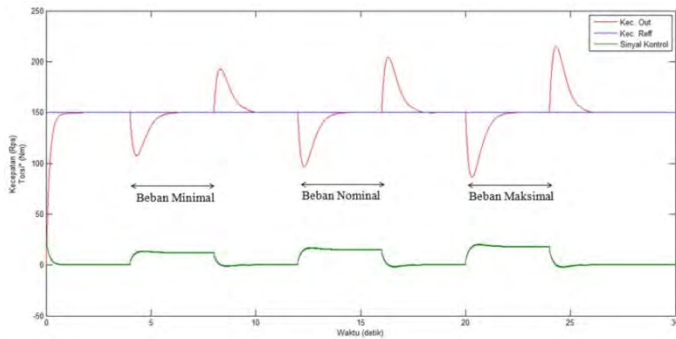
Respon torsi menunjukkan pada saat awal, torsi yang dihasilkan mengejar nilai referensi yang diberikan namun memiliki *ripple* sebesar 2 Nm. Pada waktu sekitar 0,018 detik torsi yang dihasilkan mengikuti nilai referensi torsi yang diberikan dengan nilai 23 Nm. Pada kontroler JST RBF – PD Hybrid, torsi referensi yang dihasilkan cenderung berubah lebih kasar dibandingkan kontroler PI. Hal tersebut dapat terjadi karena nilai integrator selalu berubah untuk memperkecil nilai error pada respon kecepatan yang dihasilkan.

4.3.3 Pengujian dengan Beban

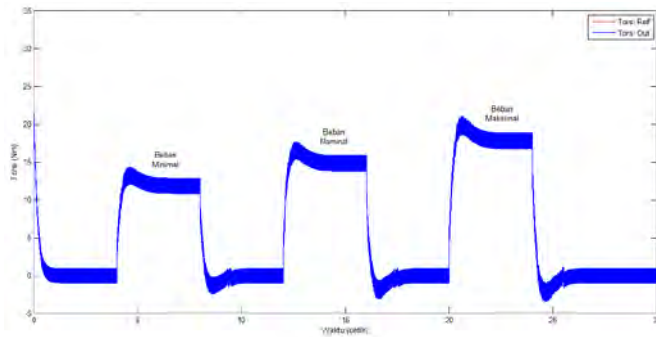
Pada bagian ini pengujian yang dilakukan sama dengan pengujian dengan beban pada kontroler PI, yaitu pengujian dilakukan dengan 2 cara yaitu beban secara langsung dan *gradual*. Representasi beban yang diberikan sesuai dengan Gambar 4.5 dan 4.8.

4.3.3.1 Beban secara langsung

Respon yang dihasilkan dari pengujian menggunakan beban secara langsung pada simulasi vektor kontrol dengan kontroler JST RBF-PD Hybrid ditunjukkan pada Gambar 4.15 dan Gambar 4.16.



Gambar 4.15 Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban langsung



Gambar 4.16 Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban langsung

Berdasarkan hasil pengujian, ketika diberi beban terjadi penurunan kecepatan untuk setiap jenis beban yang diberikan. Setelah kecepatan turun, pada waktu $t = 4,27$ detik (minimal), $t = 12,28$ detik (nominal), $t = 20,34$ detik (maksimal) kecepatan mulai naik mengejar nilai referensi dan mulai masuk pada keadaan tunak rata-rata dalam selang waktu (*settling time*) 2,68 detik dihitung dari pada saat beban mulai diberikan. Karakteristik respon kecepatan saat diberi beban dapat ditunjukkan pada Tabel 4.4.

Tabel 4.4 Karakteristik kecepatan motor induksi saat pembebanan menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid

| Karakteristik | Minimal | Nominal | Maksimal |
|-------------------------------|---------|---------|----------|
| <i>Settling time</i> (detik) | 2,60 | 2,65 | 2,80 |
| <i>Overshoot</i> (%) | 0,00 | 0,00 | 0,00 |
| <i>Error Steady state</i> (%) | 0,00 | 0,00 | 0,00 |

Kemudian setelah beban dilepas kondisi yang terjadi ialah berbanding terbalik dengan kondisi saat diberi beban yaitu kecepatan mengalami kenaikan hingga melebihi nilai referensinya. Setelah kecepatan naik, pada waktu pada waktu $t = 8,29$ detik (beban minimal), $t = 16,32$ detik (beban nominal), dan $t = 24,31$ detik (beban maksimal) kecepatan mulai turun mengejar nilai referensi dan mulai masuk pada keadaan tunak rata-rata dalam selang waktu 3,18 detik. Karakteristik respon kecepatan saat beban dilepas dapat ditunjukkan pada Tabel 4.5.

Tabel 4.5 Karakteristik kecepatan motor induksi saat lepas beban menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid

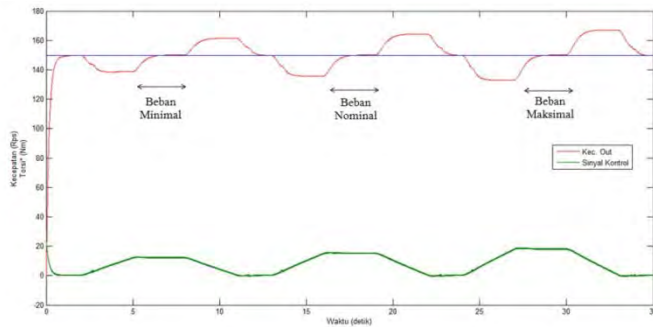
| Karakteristik | Minimal | Nominal | Maksimal |
|-------------------------------|---------|---------|----------|
| <i>Settling time</i> (detik) | 3,10 | 3,20 | 3,25 |
| <i>Undershoot</i> (%) | 0,13 | 0,13 | 0,13 |
| <i>Error Steady state</i> (%) | 0,00 | 0,00 | 0,00 |

Pada Gambar 4.16 dapat dilihat hubungan antara sinyal kontrol dengan respon kecepatan bahwa ketika sinyal kontrol mencapai nilai beban yang diberikan kecepatan masuk pada keadaan tunaknya. Lama waktu saat mulai diberi beban hingga sinyal kontrol mencapai nilai torsi beban ialah sama dengan lama waktu penurunan kecepatan yang terjadi dan begitu pula saat beban di lepas.

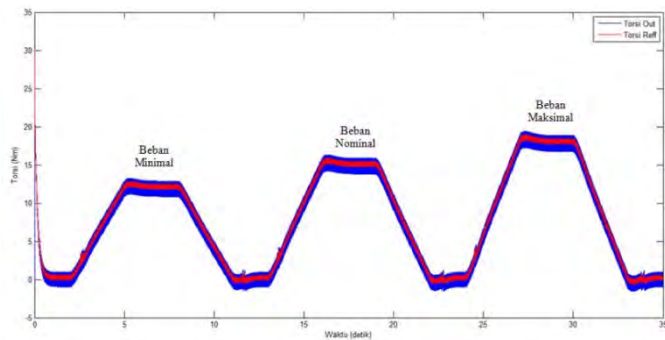
Respon torsi pada Gambar 4.17 menunjukkan pada saat terjadi perubahan beban, torsi mengikuti nilai referensi yang diberikan, namun terdapat *ripple* sebesar 2 Nm.

4.3.3.2 *Beban secara gradual*

Respon yang dihasilkan dari pengujian menggunakan beban secara *gradual* pada simulasi vektor kontrol dengan kontroler JST RBF – PD Hybrid ditunjukkan pada Gambar 4.17 dan Gambar 4.18.



Gambar 4.17 Respon kecepatan motor induksi dan sinyal kontrol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban *gradual*



Gambar 4.18 Respon torsi motor induksi menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid dan beban *gradual*

Berdasarkan hasil pengujian, ketika beban diberikan secara bertahap dari beban = 0 sampai dengan nilai beban yang ditentukan (minimal, nominal dan maksimal) terjadi penurunan kecepatan, kemudian *steady* dengan waktu rata-rata selama 2 detik. Kemudian ketika beban yang diberikan *steady*, kecepatan perlahan naik dengan rata-rata waktu untuk mencapai keadaan tunak ialah selama 2,3 detik dihitung saat beban mulai *steady*. Karakteristik respon kecepatan saat beban konstan dapat ditunjukkan pada Tabel 4.6.

Tabel 4.6 Karakteristik kecepatan motor induksi saat beban konstan menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid

| Karakteristik | Minimal | Nominal | Maksimal |
|-------------------------------|---------|---------|----------|
| <i>Settling time</i> (detik) | 2,20 | 2,30 | 2,50 |
| <i>Overshoot</i> (%) | 0,00 | 0,00 | 0,00 |
| <i>Error Steady state</i> (%) | 0,00 | 0,00 | 0,00 |

Ketika beban diturunkan terjadi kenaikan kecepatan melebihi nilai referensinya dan ketika beban ditahan pada nilai 0 kecepatan mulai turun menuju kondisi tunaknya dalam waktu rata-rata sekitar 1,88 detik. Karakteristik respon kecepatan saat beban = 0 dapat ditunjukkan pada Tabel 4.7.

Tabel 4.7 Karakteristik kecepatan motor induksi saat beban bernilai nol menggunakan vektor kontrol dengan kontroler JST RBF – PD Hybrid

| Karakteristik | Minimal | Nominal | Maksimal |
|-------------------------------|---------|---------|----------|
| <i>Settling time</i> (detik) | 1,80 | 1,85 | 2,00 |
| <i>Undershoot</i> (%) | 0,06 | 0,00 | 0,13 |
| <i>Error Steady state</i> (%) | 0,00 | 0,00 | 0,00 |

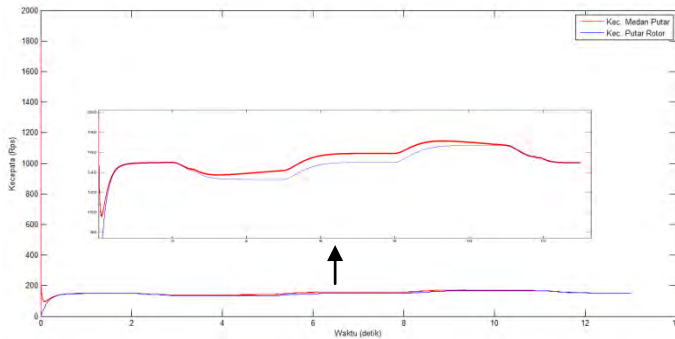
Pada Gambar 4.9 dapat dilihat hubungan antara sinyal kontrol dengan respon kecepatan, ketika sinyal kontrol naik secara *gradual* kecepatan turun dan *steady* pada nilai tertentu begitu pula saat beban *gradual* turun kecepatan naik dan *steady* pada nilai tertentu. Namun pada saat sinyal kontrol *steady* kecepatan mengengar nilai referensinya.

Jika dibandingkan dengan pengujian dengan beban langsung, beban *gradual* mengalami penurunan dan kenaikan kecepatan lebih kecil terhadap kecepatan referensinya. Respon torsi pada Gambar 4.18 menunjukkan pada saat terjadi perubahan beban nilai torsi yang dihasilkan mengikuti torsi mengikuti nilai referensi yang diberikan, namun terdapat *ripple* sebesar 2 Nm.

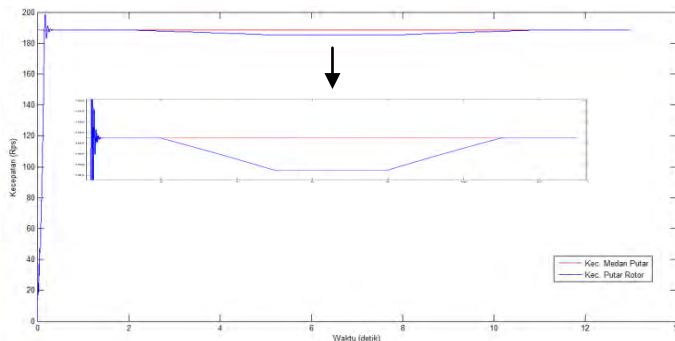
4.4 Pengamatan Nilai N_s dan N_r

Pada bagian ini dilakukan pengamatan terhadap nilai kecepatan medan putar (N_s) dan kecepatan putar rotor (N_r) dengan tujuan untuk melihat karakteristik dari metode vektor kontrol yang digunakan. Hasil pengamatan ditunjukkan pada Gambar 4.19. Sebagai pembanding

dilakukan pula pengamatan nilai N_s dan N_r tanpa menggunakan vektor kontrol seperti ditunjukkan pada Gambar 4.20.



Gambar 4.19 Pengamatan nilai nilai N_s dan N_r dengan vektor kontrol



Gambar 4.20 Pengamatan nilai nilai N_s dan N_r tanpa vektor kontrol

Berdasarkan Gambar 4.19 dapat dilihat bahwa N_s mengalami kenaikan secara drastis pada $t = 0$ lalu pada $t = 0,1$ detik N_s mulai mengikuti perubahan N_r dalam kondisi tanpa beban ($t = 0$ s/d $t = 2$ detik dan $t = 11$ s/d $t = 13$ detik) maupun berbeban ($t = 2$ s/d $t = 11$ detik). Berbeda halnya dengan Gambar 4.20, N_s bernilai konstan dari $t = 0$ sampai dengan waktu pengamatan berakhir ($t = 13$ detik) meskipun N_r berubah.

[Halaman ini sengaja dikosongkan]

BAB V

PENUTUP

5.1 Kesimpulan

Dari Tugas Akhir ini, terdapat beberapa hal yang dapat disimpulkan, yaitu :

- a. Metode vektor kontrol atau *field oriented control* memiliki karakteristik kecepatan medan putar (N_s) yang selalu mengikuti perubahan kecepatan putar rotor (N_r) di mana beban yang diberikan berbanding lurus dengan kecepatan slip yang dihasilkan.
- b. Penggunaan JST RBF – PD Hybrid sebagai kontroler kecepatan mampu mengatasi perubahan parameter beban pada motor induksi di mana pada beban langsung waktu yang dibutuhkan untuk mencapai keadaan tunak ialah rata-rata sebesar 2,68 detik saat pembebanan dan rata-rata sebesar 3,18 detik saat lepas beban dengan *undershoot* rata-rata 0,13% .
- c. Sedangkan pada beban *gradual* waktu yang dibutuhkan untuk mencapai keadaan tunak ialah rata-rata sebesar 2,3 detik saat beban konstan (minimal, nominal, maksimal) dan rata-rata sebesar 1,88 detik saat beban = 0 dengan *undershoot* rata-rata 0,06%.
- d. Respon torsi yang dihasilkan dapat mengikuti torsi referensi yang diberikan oleh kontroler, baik pada kondisi berbeban maupun tanpa beban namun memiliki *ripple* sebesar 2 Nm.

5.2 Saran

Saran yang dapat diberikan untuk Tugas Akhir ini adalah pada penelitian lanjutan untuk penentuan nilai fluks referensi agar dibuat berdasarkan *field weakening* (tidak dibuat konstan). Selain itu diharapkan kedepannya hasil simulasi dapat lebih diperbaiki dan dapat diimplementasikan pada *plant* riil.

[Halaman ini sengaja dikosongkan]

LAMPIRAN A

A. Listing Program JST-RBF

```
function uk=Kontroler_JST_RBF(xin)
global tt ek e_mod w1 w2 woh lamda_o alpha_o
alpha_1 alpha_2 jin xx u_pd t_samp
ek= xin(1);
e_mod = xin(2);
tt = xin(3); % waktu (untuk mengambil data
ke- )
u_pd = xin(4);

xmid1=0;
xmid2=1;
% proses inisialisasi
if tt==0
    jin=2;
    xx(1)=ek;
    xx(2)=0;
    w1=[ 0.0540 5.0932]';
    w2=[ 0 5.4099]';
    woh=[ 0.1922 1]';
    lamda_o=1; % learning gain zo (output)
    alpha_o=1; % learning rate woh
    alpha_1=1; % learning rate w1
    alpha_2=1; % learning rate w2
end

for epp=1:10% eppoch = siklus belajar
% Perhitungan Forward
% Menghitung output layer input
for j=1:jin
    if j==1
        xmid=xmid1;
    else
        xmid=xmid2;
    end
    zh(j)=(w2(j)*xx(j))-(xmid*w1(j));
    uh(j)=exp(-(zh(j)^2));
end
```



```

% Menghitung output neuron (layer output)
zo=0;
for j=1:jin
    zo=zo+woh(j)*uh(j);
end
uk=lamda_o*zo; % fungsi aktivasi linear
% Menghitung eror output
er=e_mod*4.618/(3195*1.12);
% Menghitung Perambatan error
for j=1:jin
    if zo==0
        erh(j)=er/jin;
    else
        erh(j)=(woh(j)*uh(j)/zo)*er;
    end
end
% Revisi Bobot dari layer hidden ke layer
output
woh(1)=woh(1)+alpha_o*er*lamda_o*uh(1);
% Revisi bobot dari layer input ke layer
hidden (nilai tengah gaussian)
for j=1:jin
    if j==1
        xmid=xmid1;
    else
        xmid=xmid2;
    end
    w1(j)=w1(j)+alpha_1*erh(j)*(-
2*zh(j)*uh(j))*xmid;
end
% Revisi bobot dari layer input ke layer
hidden
for j=1:jin
    w2(j)=w2(j)+alpha_2*erh(j)*(-
2*zh(j)*uh(j))*xx(j);
end
end
xx(1)=ek;
xx(2)=uk;

```

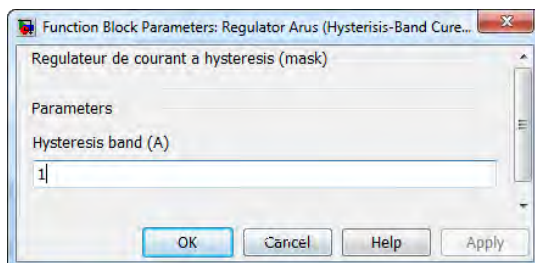
B. Listing Program Parameter Motor Induksi

```
clear all
clc
Ts=20e-6;

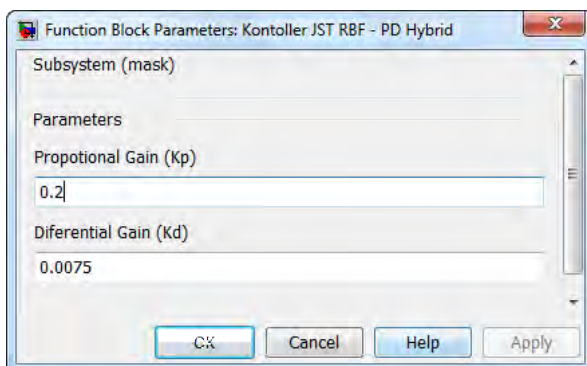
% Parameter Motor Induksi
%=====
Vd=460;
F = 60;
Ls = 0.3826;
Lr = 0.3808;
Lm = 0.3687;
Rr = 1.34;
Rs = 1.77;
P = 4;
Jr = 0.025;
Br = 0;

%Matriks B
q = [Ls 0 Lm 0; 0 Ls 0 Lm; Lm 0 Lr 0; 0 Lm 0
Lr];
qi=inv(q);
```


B. *Hysterisis Band Current Control PWM*



C. **Blok Kontroler JST RBF – PD Hybrid**



[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

- [1] Bose, B. K., *Modern Power Electronics and AC Drives*, Englewood Cliffs, New Jersey: Prentice-Hall, 2001.
- [2]. Pham Thuong Cat, Le Hung Linh, dan Minhtuan Pham. " *Speed Control of 3-Phase Asynchronous Motor Using Artificial Neural Network*".IEEE International Conference on Control and Automation.June 9-11,2010.
- [3]. Muhaimin,Moch. 2010. "*Perancangan Sistem Pengendalian Kecepatan Motor Induksi Menggunakan Kontroller Neuro PID*". Surabaya: Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember.
- [4] Zuhal, *Dasar Teknik Tenaga Listrik Dan Elektronika Daya*,Jakarta:PT Gramedia Pustaka Utama, Juni 2000.
- [5]. Kurniawan,Panji. 2010. "*Perancangan dan Simulasi Metode Direct Torque Control (DTC) untuk Motor Induksi Tiga Fasa*". Surabaya: Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember.
- [6] S. Buja, Giuseppe dan Kazmierkowski, Marian P, "*Direct Torque Control of PWM Inverter-Fed AC Motor-A Survey*",IEEE, 2004
- [7] Gamayatri, Nurlita, *Diktat Dasar Sistem Pengaturan*, Surabaya: Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember.
- [8] Mohan, Ned, *Advance Electric Drives Analysis, Control and Modeling using Simulink*, MNPERE,2001
- [9] Ong, Che-Mun, *Dynamic Simulation Of Electric Machinery using Matlab/Simulink*, , New Jersey: Prentice-Hall.

[Halaman ini sengaja dikosongkan]

RIWAYAT HIDUP



Rizalun Iqbal, lahir di Bandung pada tanggal 11 Oktober 1990. Putra pertama dari pasangan ayahanda Dede Ibrahim dan ibunda Seli Apriyati. Setelah menempuh pendidikan formal di SDN Guruminda Bandung, SMP Negeri 13 Bandung dan SMA Negeri 12 Bandung, penulis melanjutkan melanjutkan studi Diploma 3 jurusan Teknik Otomasi Manufaktur dan Mekatronika di Politeknik

Manufaktur Bandung dan lulus tahun 2011. Kemudian melanjutkan kuliah Lintas Jalur Sarjana di Institut Teknologi Sepuluh Nopember Surabaya dengan mengambil Jurusan Teknik Elektro, Bidang Studi Teknik Sistem Pengaturan. Pada akhir masa pendidikannya, penulis mengerjakan tugas akhir di bawah bimbingan Dr. Ir. Mochammad Rameli dan Ir. Rusdhianto Effendi A.K, M.T, dengan judul Pengaturan Kecepatan Motor Induksi 3 Fasa Menggunakan Metode Vektor Kontrol Berbasis Jaringan Syaraf Tiruan *Radial Basis Function* – PD (JST RBF – PD) *Hybrid*.

[Halaman ini sengaja dikosongkan]